Markov chain Monte Carlo

Generating variates from a probability distribution $p(x)$ is not straightforward when

- x is high-dimensional (but not Gaussian),
- $p(x)$ is of non-standard form.

These difficulties occur often (but not exclusively) in Bayesian inference.

We seek a way of simulating from $p(x)$ in this situation.

A brief introduction to Markov chains

A *Markov chain* is a countable sequence of random variables, $\{X_t\}$, for which

$$\mathbb{P}[X_t \in B | X_1 = x_1, \ldots, X_{t-1} = x_{t-1}] = \mathbb{P}[X_t \in B | X_{t-1} = x_{t-1}]$$

that is, X_t is conditionally independent of $X_0, \ldots, X_{t-2}$ given X_{t-1}.

In the simplest case, the $\{X_t\}$ takes values on the finite (discrete) state space

$$\mathcal{S}_X = \{s_1, \ldots, s_d\}$$

and the chain is *homogeneous*, that is, the stochastic properties of $\{X_t\}$ do not change with time.

A brief introduction to Markov chains

A homogeneous, discrete state-space Markov chain is characterized by its initial state X_0, and its *transition matrix* P, a $d \times d$ *stochastic* matrix whose rows sum to one, such that

$$P_{ij} = \mathbb{P}[X_t = s_j | X_{t-1} = s_i]$$

describes the entire set of one-step ahead conditional probabilities. Denote by

$$p_{ij}(k) = \mathbb{P}[X_{t_0+k} = s_j | X_{t_0} = s_i]$$

the k-step ahead probabilities. Note that if $P(k)$ is the matrix of k-step ahead transition probabilities, then

$$P(k) = P^k$$

A brief introduction to Markov chains

The properties of the chain depend on P. The chain is

- *irreducible* if $p_{ij}(k) > 0$, for all i, j, and at least one k.
- *aperiodic* if all states have period 1: that is, for each i, returns to state i can occur after any number of steps. The *period* of state i is defined as the greatest common divisor of the set of possible return times, $\mathcal{R}$,

$$\mathcal{R} = \{r : \mathbb{P}[X_r = s_i | X_0 = s_i] > 0\}$$

- *recurrent* if all states are recurrent, that is, the probability of returning to each state in a finite number of steps is positive. Let $T_i = \inf\{k : X_k = i | X_0 = i\}$. State i is recurrent if and only if

$$\mathbb{P}[T_i = \infty] = 0$$

and *transient* otherwise. If $\mathbb{E}[T_i] < \infty$, state i is termed *positive recurrent*, otherwise it is termed *null recurrent*.

A brief introduction to Markov chains

The *stationary* or *equilibrium distribution*, π, of a homogeneous Markov chain is the $1 \times d$ vector of probabilities such that

$$\pi = \pi P$$

that is, for each i,

$$\pi_i = \sum_j \pi_j P_{ji}$$

An irreducible chain has a stationary distribution if and only if all of its states are positive recurrent, in which case π is unique. Given P, π can be computed as

$$\lim_{k \longrightarrow \infty} P^k = \mathbf{1}\pi.$$

where $\mathbf{1}$ is the $d \times 1$ vector of 1s. A matrix P with the stationary distribution π can be computed by solving the system of equations $\pi = \pi P$.

Key aspects of the stationary distribution are that

(a) As the k-step ahead probability matrix P^k converges to a matrix with d identical rows, the Markov chain can eventually "forget" its initial value X_0.

(b) Realizations of $\{X_t\}$ exhibit statistical properties that demonstrate (strong) convergence to the stationary distribution, that is, for $i = 1, \ldots, d$,

$$\lim_{n \longrightarrow \infty} \frac{\sum_{k=1}^{n} \mathbb{1}_{\{s_i\}}(X_k)}{n} = \pi_i$$

A homogeneous Markov chain with stationary distribution π is *reversible* if

$$\pi_i P_{ij} = \pi_j P_{ji}$$

for all states i and j. This is also termed the *detailed balance* condition.

Note that **if** this equation holds for a specified π, **then** this implies that the P has been specified so as to have stationary distribution π.

Note that the $\{X_t\}$ are **dependent** random variables, so the standard frequentist asymptotic laws do not directly apply.

However, the *ergodic theorem* applies for irreducible, aperiodic and positive recurrent Markov chains, in particular

$$\frac{1}{N}\sum_{t=1}^{N} g(X_t) \xrightarrow{a.s.} \mathbb{E}_\pi[g(X)]$$

for all bounded functions g, provided

$$\mathbb{E}_\pi[|g(X)|] < \infty$$

A Central Limit Theorem result also holds under mild regularity conditions, specifically,

$$\sqrt{N}\left(\frac{1}{N}\sum_{t=1}^{N} g(X_t) - \mathbb{E}_\pi[g(X)]\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2(g))$$

where

$$\sigma^2(g) = \mathbb{V}ar_\pi[g(X_0)] + 2\sum_{t=1}^{\infty} \mathbb{C}ov_\pi[g(X_0), g(X_t)].$$

Discrete Markov chain: example

Example ($d = 2$)

Consider $d = 2$, with

$$P = \begin{bmatrix} 0.3 & 0.7 \\ 0.9 & 0.1 \end{bmatrix}$$

Then $\pi = (9/16, 7/16)$. Here

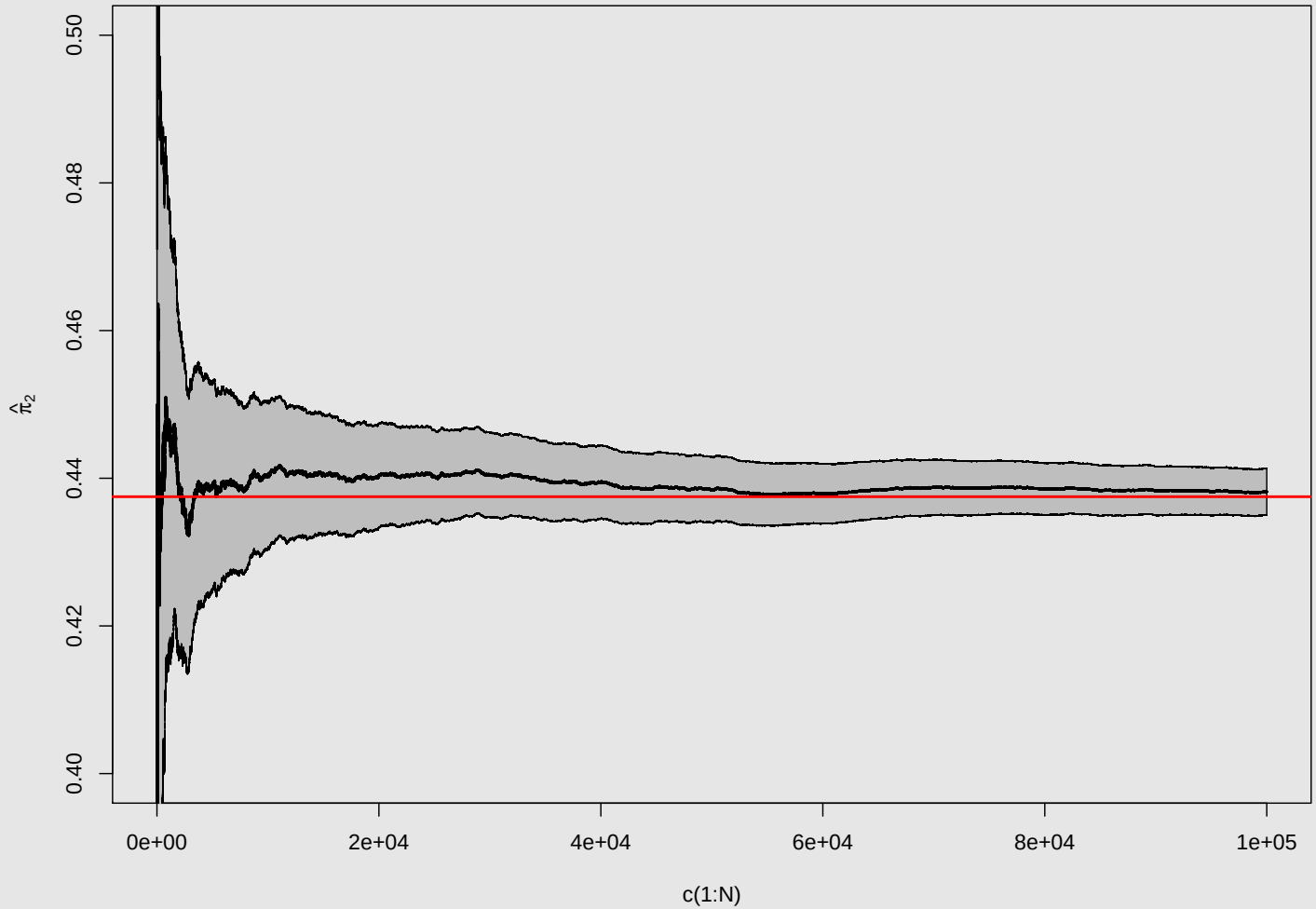
$$\pi_1 P_{12} = \frac{9}{16} \times \frac{7}{10} = \frac{63}{160} \qquad \pi_2 P_{21} = \frac{7}{16} \times \frac{9}{10} = \frac{63}{160}$$

so this chain is reversible.

Discrete Markov chain: example

Example ($d = 2$)

Sample average relative frequency of being in state 2 over 100000 steps.



Constructing reversible chains

In the 2×2, for a reversible chain, we require

$$\pi_1 P_{12} = (1 - \pi_1) P_{21}$$

Suppose

$$\begin{aligned} P_{12} &= \min\left\{1, \frac{1-\pi_1}{\pi_1}\right\} \\ P_{21} &= \min\left\{1, \frac{\pi_1}{1-\pi_1}\right\} \end{aligned}$$

Then

$$\begin{aligned} \pi_1 P_{12} &= \pi \min\left\{1, \frac{\pi_1}{1-\pi_1}\right\} = \min\left\{\pi_1, 1-\pi_1\right\} = \min\left\{1-\pi_1, \pi_1\right\} \\ &= (1-\pi_1) \min\left\{1, \frac{\pi_1}{1-\pi_1}\right\} = (1-\pi_1) P_{21} \end{aligned}$$

Constructing reversible chains

The above Markov chain we can think of as acting as follows:

- **If** $X_t = 1$: propose setting $X_{t+1} = 2$, but only accept this move with probability

$$\min\left\{1, \frac{1-\pi_1}{\pi_1}\right\}$$

  otherwise set $X_{t+1} = 1$.
- **If** $X_t = 2$: propose setting $X_{t+1} = 1$, but only accept this move with probability

$$\min\left\{1, \frac{\pi_1}{1-\pi_1}\right\}$$

  otherwise set $X_{t+1} = 2$.

Constructing reversible chains

A generalization of this approach is as follows:

- If $X_t = 1$, simulate Z_t on the set $\{1, 2\}$ with probabilities (q_{11}, q_{12}).
- If $X_t = 2$, simulate Z_t on the set $\{1, 2\}$ with probabilities (q_{21}, q_{22}).

Constructing reversible chains

- If $X_t = 1$, if $Z_t = 2$, set $X_{t+1} = Z_t = 2$ with probability

$$\alpha_{12} = \min\left\{1, \frac{\pi_2}{\pi_1}\frac{q_{21}}{q_{12}}\right\}$$

  otherwise set $X_{t+1} = X_t = 1$.

- If $X_t = 2$, if $Z_t = 1$, set $X_{t+1} = Z_t = 1$ with probability

$$\alpha_{21} = \min\left\{1, \frac{\pi_1}{\pi_2}\frac{q_{12}}{q_{21}}\right\}$$

  otherwise set $X_{t+1} = X_t = 2$.

Constructing reversible chains

In this case, we have the transition probabilities given as follows:

$$\begin{aligned} P_{12} &= \mathbb{P}[X_{t+1} = 2 | X_t = 1] = q_{12}\alpha_{12} \\ P_{21} &= \mathbb{P}[X_{t+1} = 1 | X_t = 2] = q_{21}\alpha_{21} \end{aligned}$$

to therefore

$$\pi_1 P_{12} = \pi_1 q_{12}\alpha_{12} = \pi_1 q_{12} \min\left\{1, \frac{\pi_2}{\pi_1}\frac{q_{21}}{q_{12}}\right\} = \min\{\pi_1 q_{12}, \pi_2 q_{21}\}$$

$$\pi_2 P_{21} = \pi_2 q_{21}\alpha_{21} = \pi_2 q_{21} \min\left\{1, \frac{\pi_1}{\pi_2}\frac{q_{12}}{q_{21}}\right\} = \min\{\pi_2 q_{21}, \pi_1 q_{12}\}$$

and

$$\pi_1 P_{12} = \pi_2 P_{21}.$$

Constructing reversible chains

This allows the generalization to the more general discrete state space $\{1, 2, \ldots, d, \ldots\}$. Suppose that π is an arbitrary discrete distribution. Let matrix Q define the *proposal probabilities*

$$[Q]_{ij} = \mathbb{P}[Z_t = j | X_t = i]$$

Define the *acceptance probabilities*

$$\alpha_{ij} = \min\left\{1, \frac{\pi_j}{\pi_i}\frac{q_{ji}}{q_{ij}}\right\}$$

If $X_t = i$, set $X_{t+1} = Z_t = j$ with probability α_{ij}. Otherwise set $X_{t+1} = X_t = i$.

Constructing reversible chains

This Markov chain satisfies detailed balance

$$\pi_i P_{ij} = \pi_j P_{ji} \qquad \text{for all } i, j$$

provided it is irreducible, aperiodic and positive recurrent.

Note that the rows of Q must sum to 1 as

$$\sum_{j=1}^{\infty} \mathbb{P}[Z_t = j | X_t = i] = 1$$

so Q defines a stochastic proposal (or *transition*) matrix.

Constructing reversible chains: example

Example (Poisson distribution)

Suppose, for $\lambda > 0$,

$$\pi_i = \frac{e^{-\lambda}\lambda^i}{i!} \qquad i = 0, 1, 2, \ldots.$$

Suppose

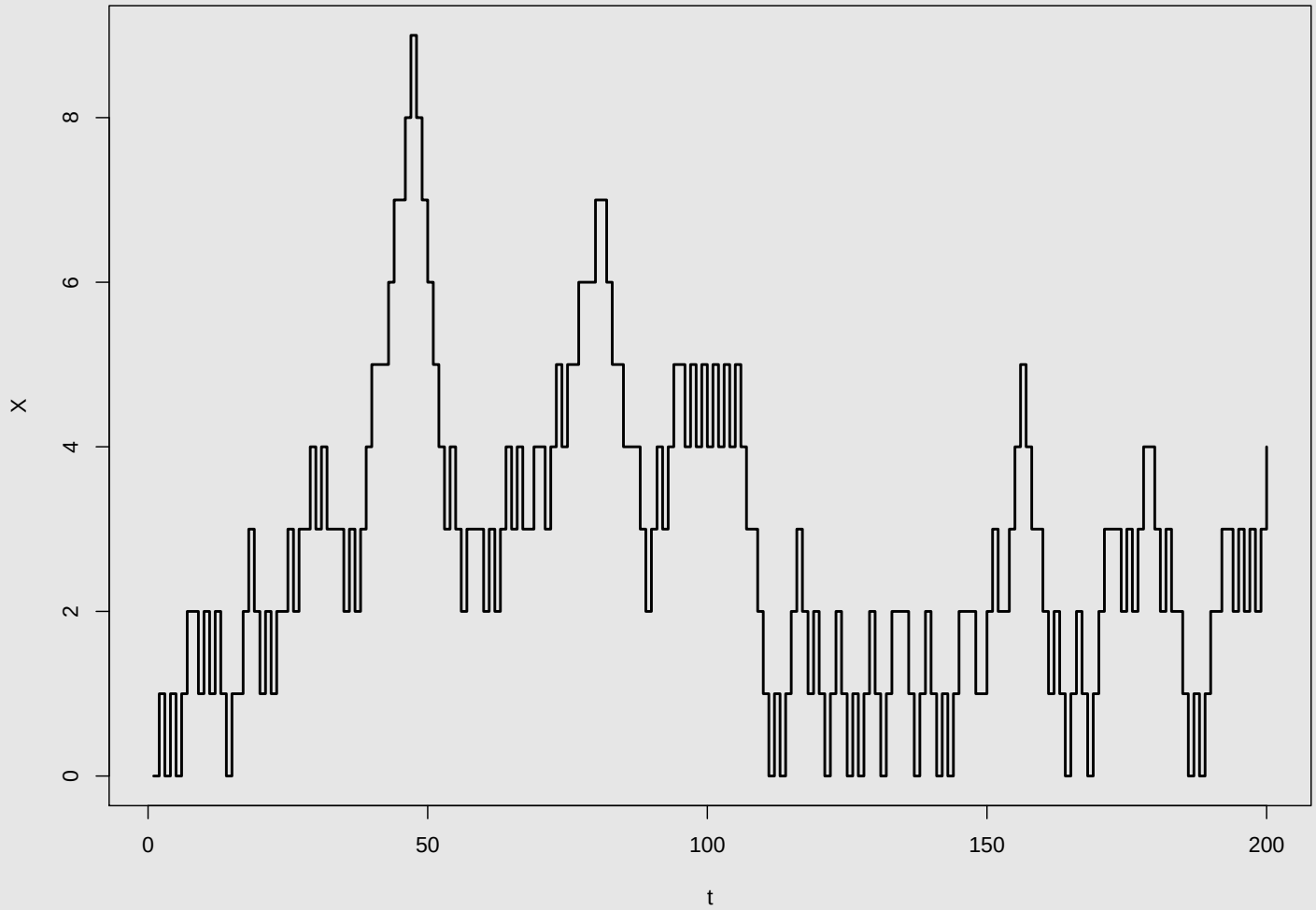
$$q_{ij} = \begin{cases} 1 & i = 0, j = 1 \\ \frac{1}{2} & i \geqslant 1, j = i-1, i+1 \\ 0 & \text{otherwise} \end{cases}$$

That is, Z_t is proposed uniformly on the finite set $\{x_t - 1, x_t + 1\}$, unless $X_t = 0$, in which case $Z_t = 1$ is proposed.

Constructing reversible chains: example

Example (Poisson distribution)

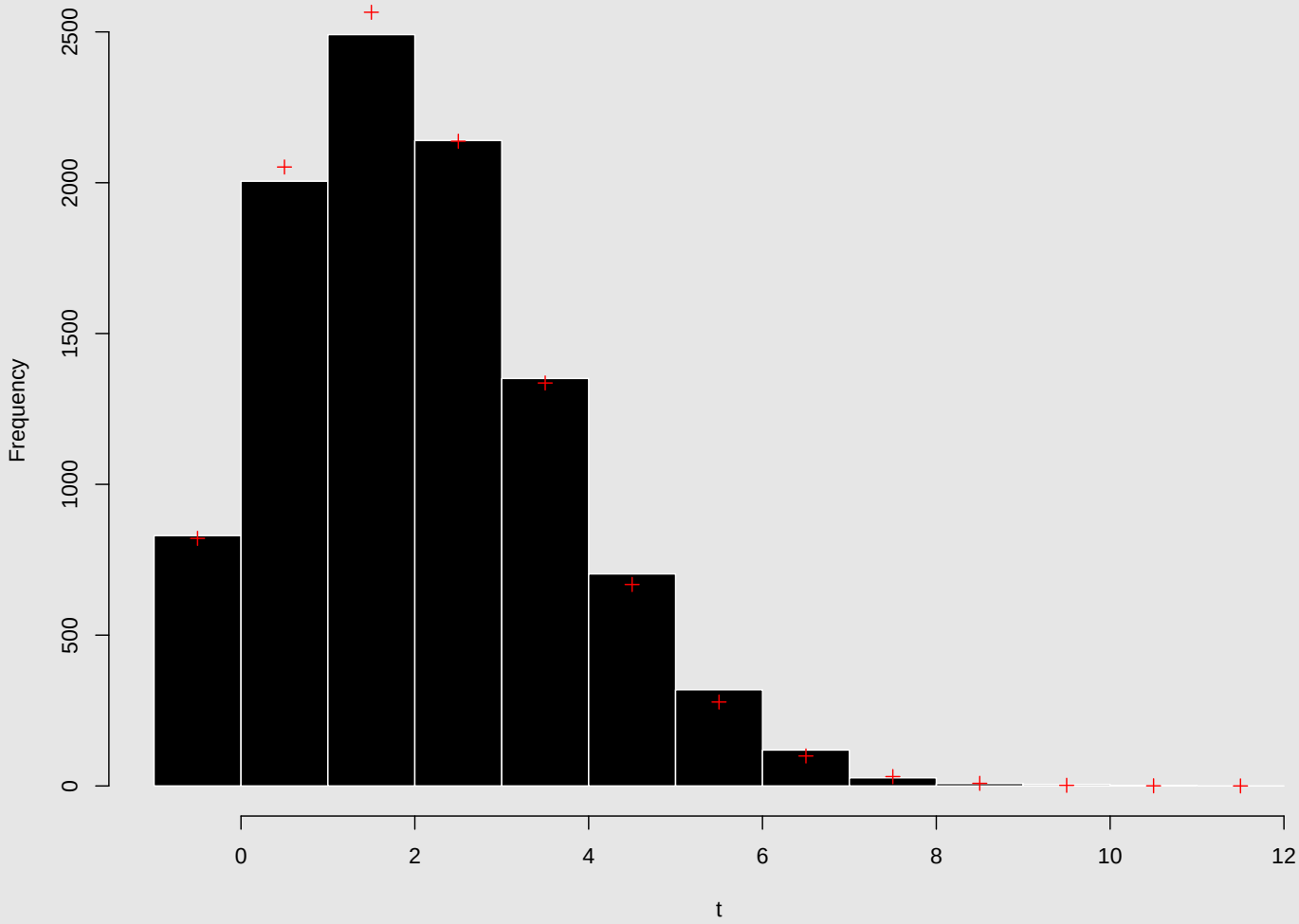
First 200 steps of the chain starting at $X_0 = 0$ with $\lambda = 2.5$.



Constructing reversible chains: example

Example (Poisson distribution)

Histogram of states visited over $N = 10000$ steps (with true values (+))



Constructing reversible chains: example

Example (Poisson distribution)

Relative frequencies ($\widehat{\pi}_i$) of states visited over $N = 10000$ steps, with true Poisson probabilities (π_i).

i	0	1	2	3	4	5
$\widehat{\pi}_i$	0.0830	0.2005	0.2491	0.2140	0.1351	0.0703
π_i	0.0821	0.2052	0.2565	0.2138	0.1336	0.0668
i	6	7	8	9	10	11
$\widehat{\pi}_i$	0.0319	0.0119	0.0027	0.0008	0.0005	0.0002
π_i	0.0278	0.0099	0.0031	0.0009	0.0002	0.0000

Constructing reversible chains: example

Example (Poisson distribution)

To measure convergence, we can use the *Total Variation* distance. For two mass functions $f_1(x)$ and $f_2(x)$, we compute

$$d_{TV}(f_1, f_2) = \frac{1}{2} \sum_x |f_1(x) - f_2(x)|$$

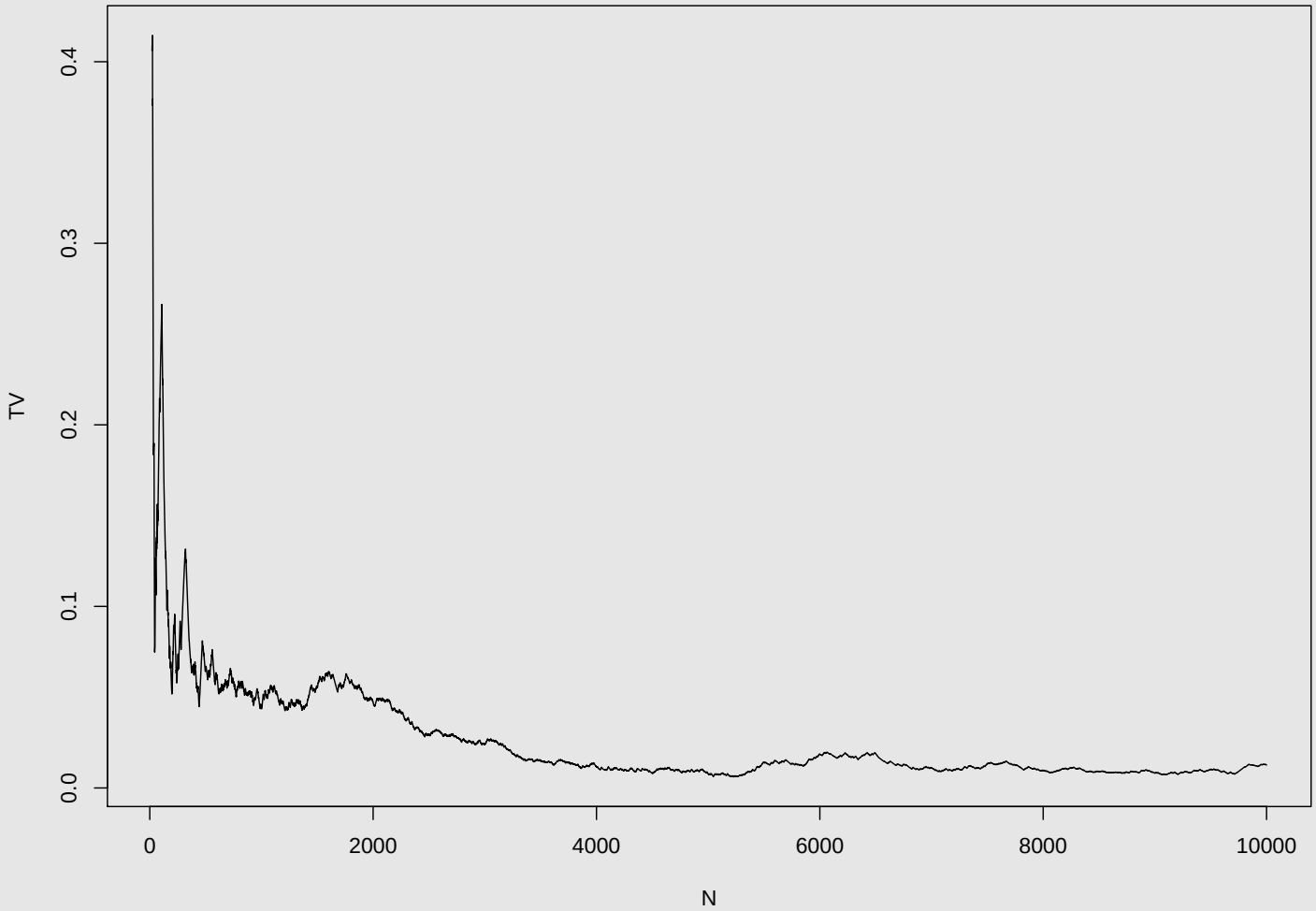
Here we take f_1 to be the true Poisson mass function π, and f_2 to be the estimated mass function $\widehat{\pi}$.

We monitor this distance as N increases.

Constructing reversible chains: example

Example (Poisson distribution)

$d_{TV}(\pi, \widehat{\pi})$ as N increases.



Measuring convergence

Note: For continuous densities (defined with respect to Lebesgue measure) the total variation distance is given by

$$d_{TV}(f_1, f_2) = \frac{1}{2} \int |f_1(x) - f_2(x)| \, dx$$

In general, the total variation distance $d_{TV}(f_1, f_2)$ can be hard to compute.

Measuring convergence

Recall the relationship between $d_{TV}(f_1, f_2)$ and the *Hellinger* distance $d_H(f_1, f_2)$

$$d_H(f_1, f_2) = \sqrt{\sum_x \left(\sqrt{f_1(x)} - \sqrt{f_2(x)}\right)^2}$$

that is,

$$\frac{1}{2} d_H^2(f_1, f_2) \leqslant d_{TV}(f_1, f_2) \leqslant d_H(f_1, f_2)$$

These inequalities also hold in the continuous case. Note that

$$d_{TV}(f_1, f_2) \longrightarrow 0 \qquad \Longleftrightarrow \qquad d_H(f_1, f_2) \longrightarrow 0.$$

The Hellinger distance is sometimes easier to compute.

Discrete Markov chains: Recap

For an arbitrary discrete distribution, π, we can now construct a Markov chain that has π as its stationary distribution, and can produce dependent samples from the π by running the Markov chain and collecting the generated states.

- specify the stochastic matrix Q
- initialize the chain by setting X_0
- for each t, if $X_t = i$, use the ith row of Q as a discrete distribution for proposing Z_t
- If $Z_t = j$, accept $X_{t+1} = j$ with probability α_{ij}

$$\alpha_{ij} = \min\left\{1, \frac{\pi_j}{\pi_i}\frac{q_{ji}}{q_{ij}}\right\}$$

otherwise set $X_{t+1} = X_t = i$.

Continuous State-Space Markov Chains

The theory above extends (reasonably straightforwardly) to continuous state spaces, that is, the countable state set $\{s_1, ..., s_d\}$ is replaced by a continuum of possible values, $\mathbb{X}$.

In this case, instead of having a transition matrix, we have a *transition kernel*

$$P(x, B) = \int_B P(x, z)\, dz$$

$P(x, B)$ determines the probability of making the transition from current value x into the set $B \subset \mathbb{X}$ in any given step.

Continuous State-Space Markov Chains

We retain the *discrete time* nature of the Markov chain, and again consider outcome sequences $\{x_1, x_2, ..., x_n, ...\}$).

Transitions are implemented using a *transition density*

$$P(x, z) \equiv P(x \to z)$$

which specifies a conditional probability density in z, given the current value x, for $x, z \in \mathbb{X}$.

Continuous State-Space Markov Chains

By analogy with the discrete case, the stationary distribution π for the continuous state space chain must satisfy

$$\pi(x) = \int P(z, x)\pi(z)\, dz$$

A reversible chain must satisfy detailed balance

$$\pi(x)P(x, z) = \pi(z)P(z, x)$$

for all x and z. Given P, we can, in theory solve for π.

In the context of sampling from probability distributions, we wish to specify π, and then find a P such that its equilibrium distribution is π.

The Metropolis-Hastings Algorithm

We attempt to mimic the construction of a Markov chain with stationary distribution π used in the discrete case.

Let Q be any proposal (transition) kernel suitable for moving (exhaustively) around $\mathbb{X}$, with associated transition density q such that

$$q(z, x) = q(z \longrightarrow x) > 0$$

for all x, z.

In fact, this can be relaxed to the condition that requires $Q^n(x, z) > 0$ for all $x, z \in \mathbb{X}$, separated by n steps in the chain).

The Metropolis-Hastings Algorithm

Then, for $z \neq x$, define

$$P(x, z) = q(x, z)\, \alpha(x, z)$$

where

$$\alpha(x, z) = \min\left\{1, \frac{\pi(z)}{\pi(x)} \frac{q(z, x)}{q(x, z)}\right\}$$

defines an *acceptance probability* for the move from x to z.

The Metropolis-Hastings Algorithm

Under this transition kernel or density P with transition density if the current state of the chain at step n is $X_n = x$, then the next value of the chain is either

- a **new value** $X_{n+1} = z$, generated from the conditional density $q(x, z)$,
- or the **current value** $X_{n+1} = x$.

The value z the proposed or *candidate* state.

Thus, starting from the n^{th} step when $x_n = x$, we have the following algorithm for implementing the continuous state space Markov chain:

The Metropolis-Hastings Algorithm

1. Generate candidate z from the conditional density $q(.,.)$ given x
2. Compute $\alpha(x, z)$
3. Generate u from $Uniform(0, 1)$
   - if $u \leqslant \alpha(x, z)$, **accept** the move to z and set $X_{n+1} = z$
   - if $u > \alpha(x, z)$, **reject** the move to z and set $X_{n+1} = x$
4. Return to 1 to generate x_{X+2}

and so on.

This is the *Metropolis-Hastings algorithm*

The Metropolis Algorithm

The general algorithm above has some special cases of interest. If q is chosen such that

$$q(x, z) = q(z, x)$$

so that q is symmetric in its arguments, then

$$\alpha(x, z) = \min\left\{1, \frac{\pi(z)}{\pi(x)}\right\}$$

and the move to z is accepted with certainty if the target probability density at z is higher than at x.

The Metropolis Algorithm

A simple symmetric transition density has

$$Z|X_n = x \sim \mathcal{N}\left(x, \sigma_q^2\right)$$

Choosing σ_q^2 small encourages many small moves.
This is the original Markov chain simulation algorithm, known as the *Metropolis Algorithm*.

Many such "local" moves can be proposed. Note that it is important to respect any parameter constraints in the proposal.

Independence Metropolis-Hastings

Another common choice is the *independence Metropolis-Hastings algorithm*, where

$$q(x, z) = q(z)$$

that is, independent of the current value of the chain. This still defines a Markov chain as

$$p(x, z) = q(z)\,\alpha(x, z)$$

still depends on x through $\alpha(x, z)$.

A good independence Markov chain (that traverses $\mathbb{X}$ quickly) is more difficult to construct without knowledge of π.

However, if π can be well-approximated by a density q (as in rejection sampling), then this method can work well.

Metropolis-Hastings algorithm

Example (Gamma density)

Suppose, for $\gamma > 0$

$$\pi(x) = \frac{1}{\Gamma(\gamma)} x^{\gamma-1} e^{-x} \qquad x > 0.$$

Suppose, first that $q(x, z)$ is specified as a *reflected* normal density, that is, we propose z by simulating

$$Y | X_t = x \sim \mathcal{N}(x, \sigma_q^2),$$

and setting $Z = |Y|$.

Metropolis-Hastings algorithm

Example (Gamma density)

Note that

$$\mathbb{P}[Z \leqslant z|X = x] = \mathbb{P}[|Y| \leqslant z|X = x] = \mathbb{P}[-z \leqslant Y \leqslant z|X = x]$$

so therefore

$$\mathbb{P}[Z \leqslant z|X = x] = \Phi((z - x)/\sigma_q) - \Phi((-z - x)/\sigma_q)$$

and, on differentiation wrt z,

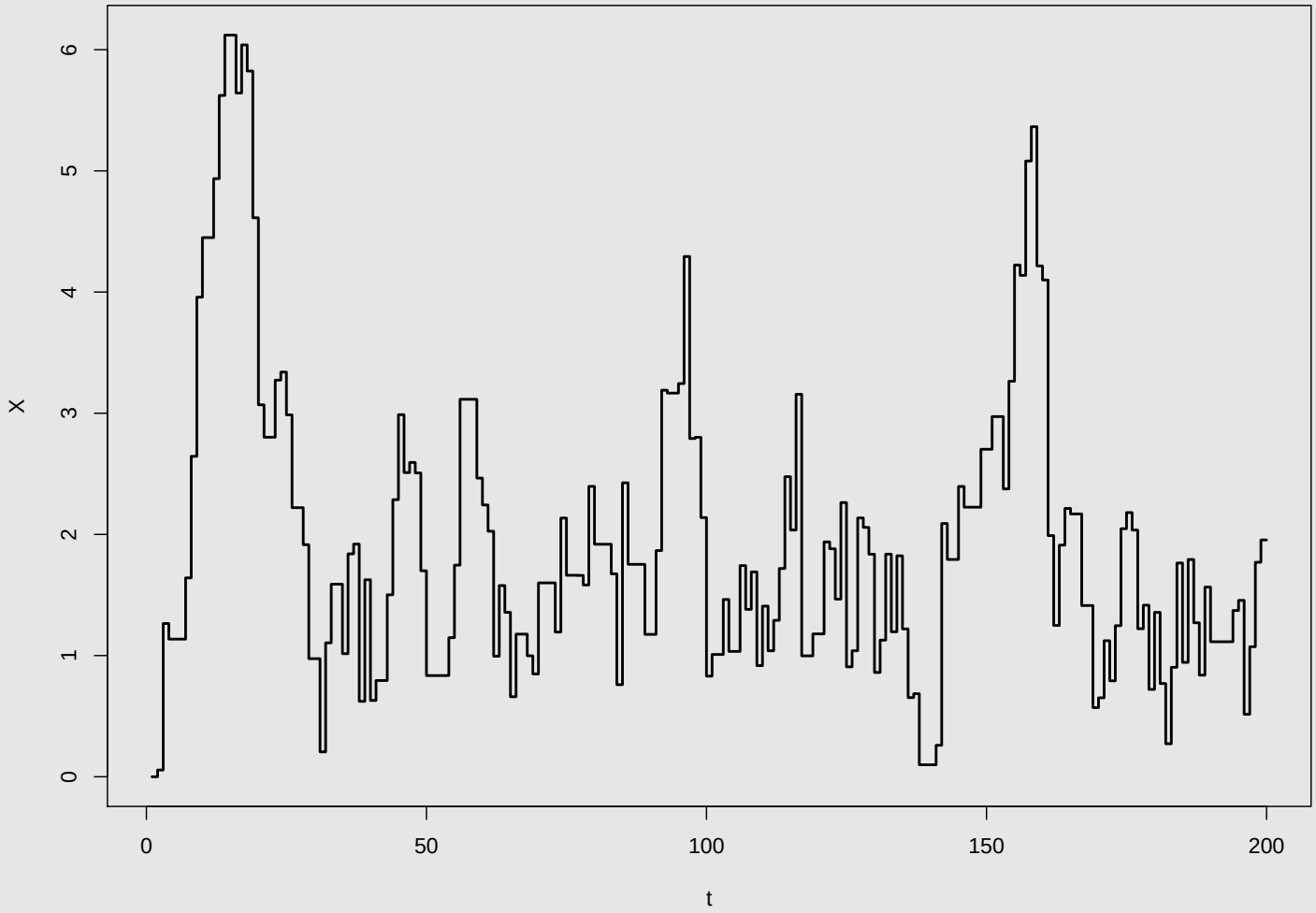
$$q(x, z) = \frac{1}{\sigma_q}(\phi((z - x)/\sigma_q) + \phi((-z - x)/\sigma_q)) = q(z, x)$$

as ϕ is an even function.

Metropolis-Hastings algorithm

Example (Gamma density)

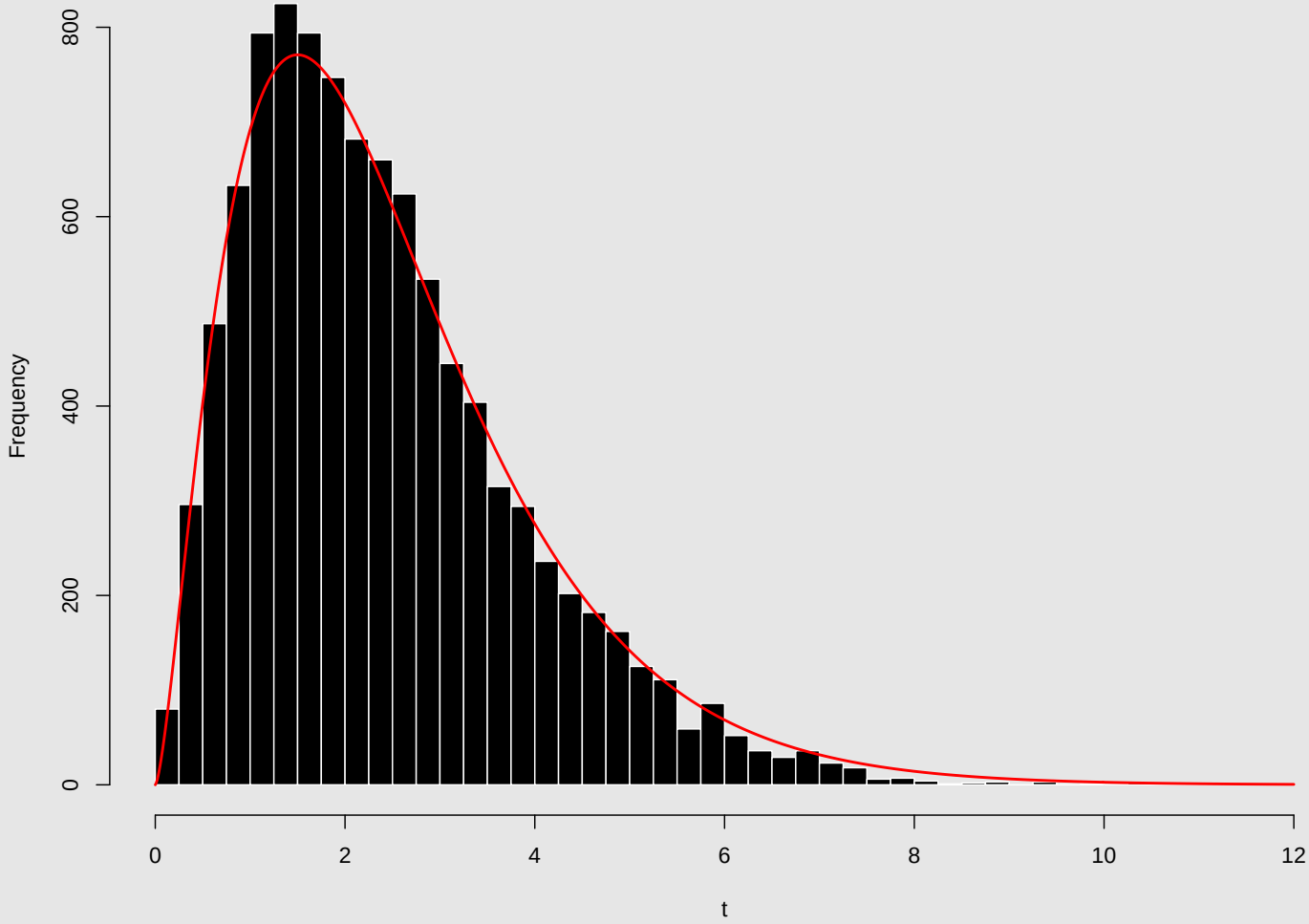
First 200 steps of the chain starting at $X_0 = 0$ with $\gamma = 2.5$, $\sigma_q = 1$.



Metropolis-Hastings algorithm

Example (Gamma density)

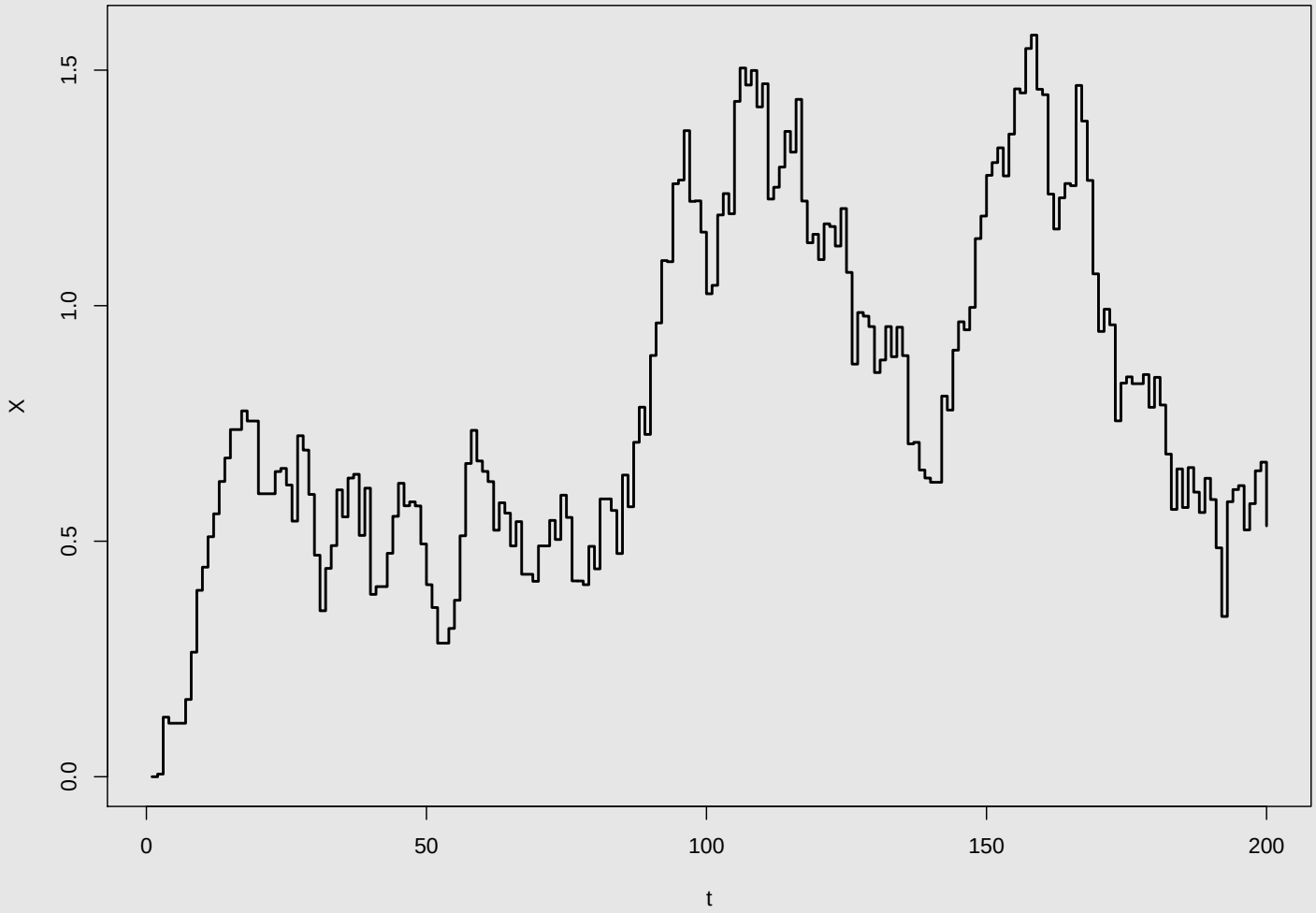
Histogram of states visited over $N = 10000$ steps (with Gamma density (solid))



Metropolis-Hastings algorithm

Example (Gamma density)

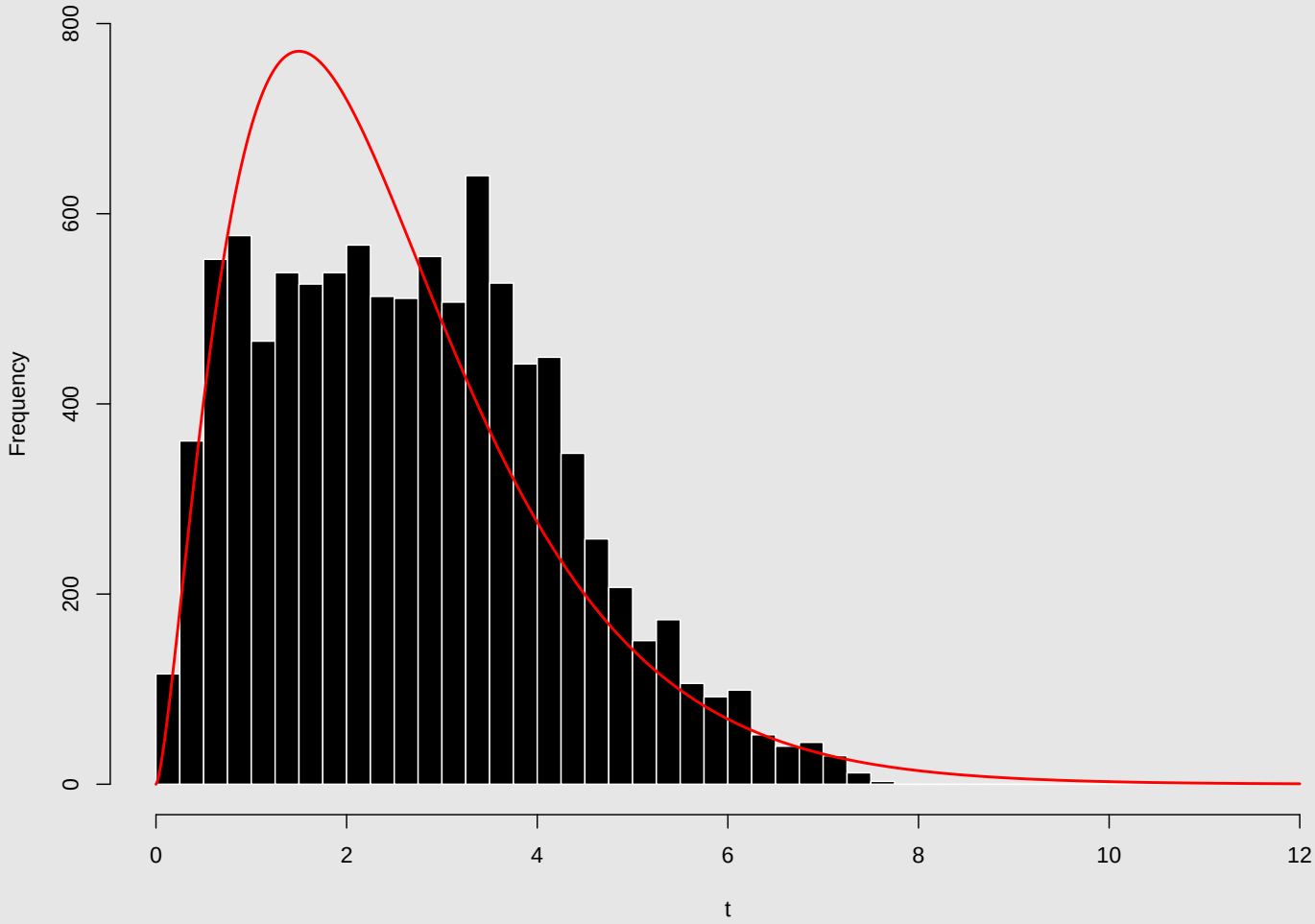
First 200 steps of the chain starting at $X_0 = 0$ with $\gamma = 2.5$, $\sigma_q = 0.1$.



Metropolis-Hastings algorithm

Example (Gamma density)

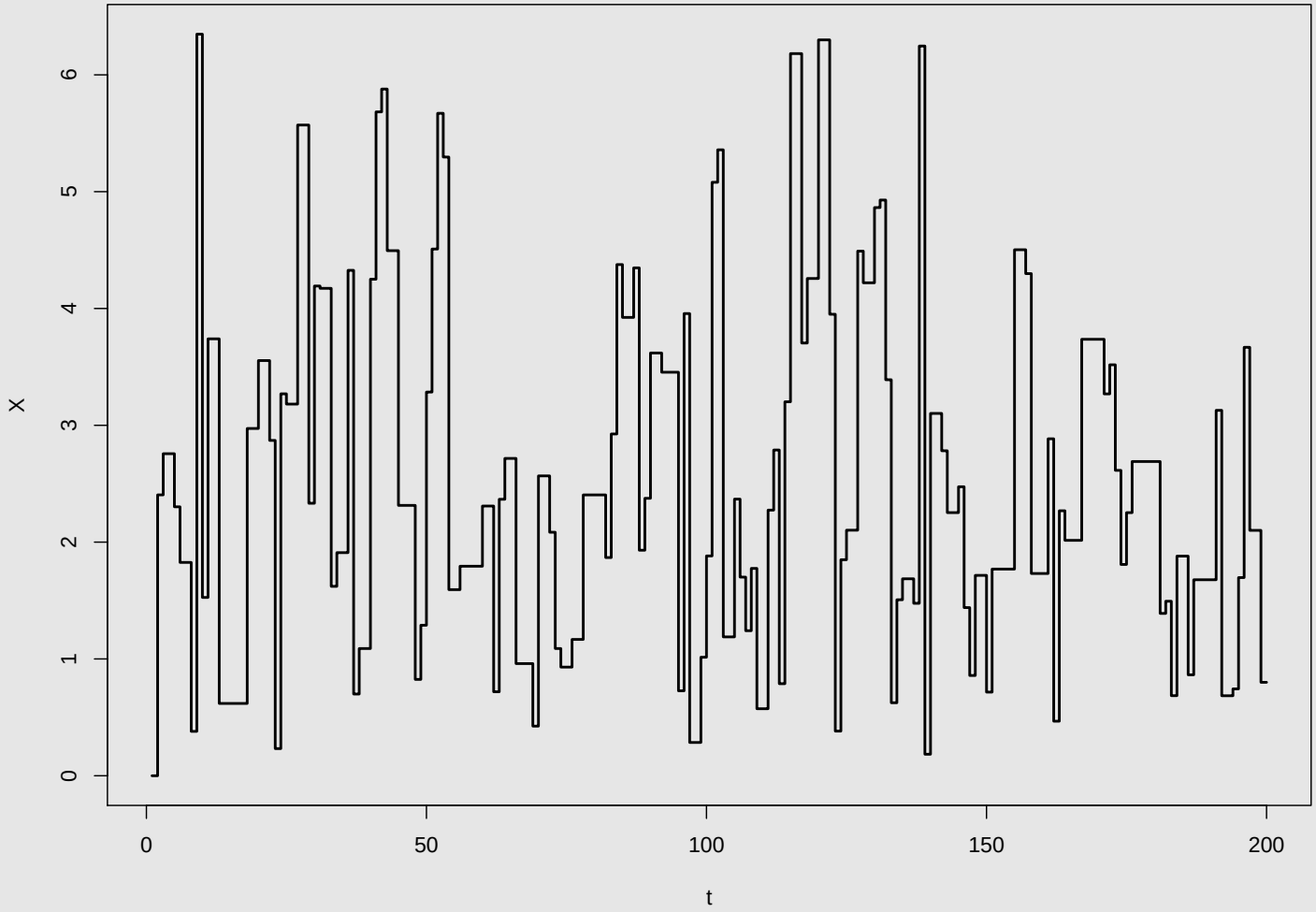
Histogram of states visited over $N = 10000$ steps (with Gamma density (solid))



Metropolis-Hastings algorithm

Example (Gamma density)

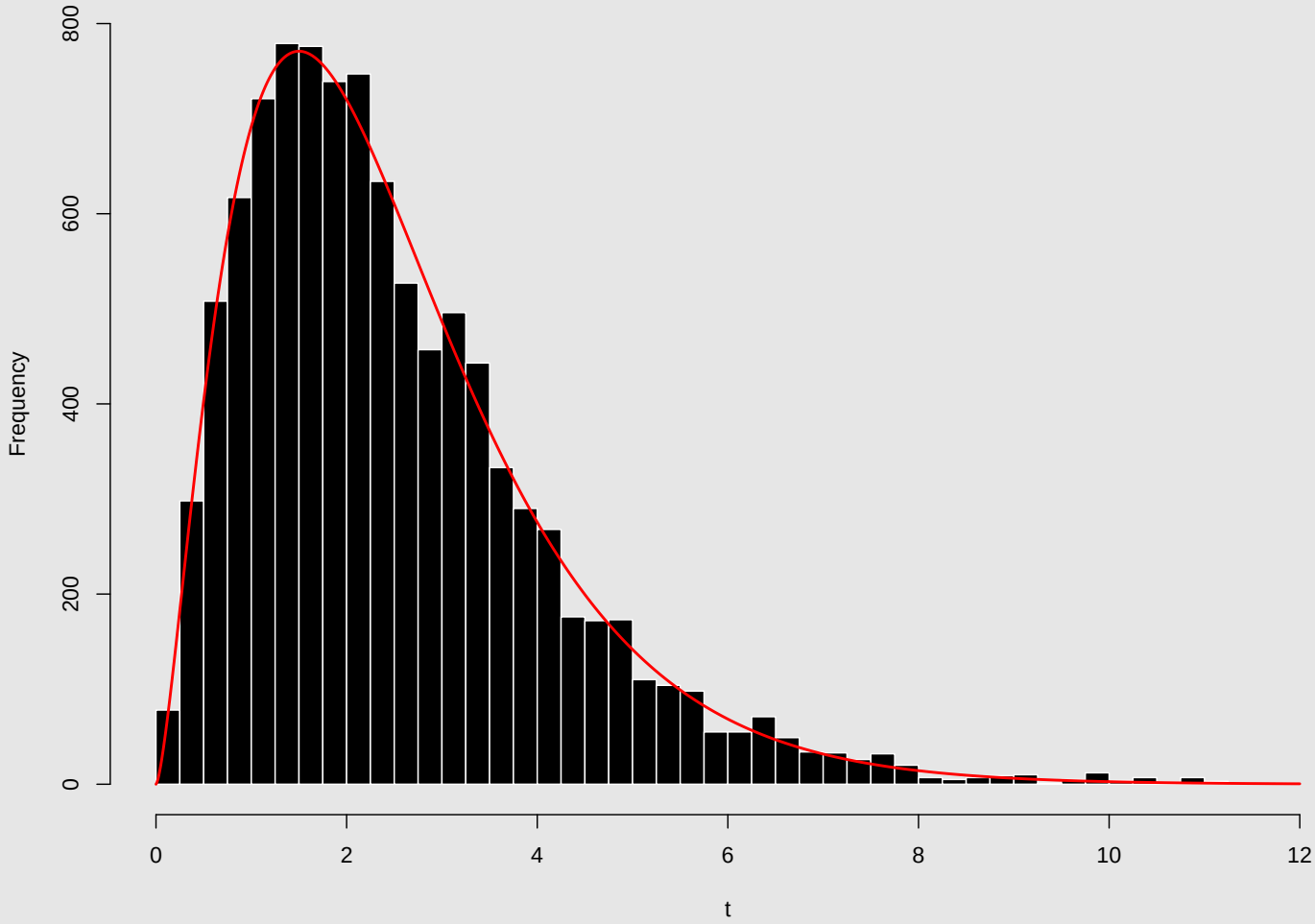
First 200 steps of the chain starting at $X_0 = 0$ with $\gamma = 2.5$, $\sigma_q = 3$.



Metropolis-Hastings algorithm

Example (Gamma density)

Histogram of states visited over $N = 10000$ steps (with Gamma density (solid))



Metropolis-Hastings algorithm

The states generated by the Markov chain are correlated; we can assess the performance of the Markov chain by examining the sample autocorrelation function

$$r(k) = \left(\frac{N-1}{N-k-1} \right) \frac{\sum_{t=k+1}^{N} (x_t - \overline{x})(x_{t-k} - \overline{x})}{\sum_{t=1}^{N} (x_t - \overline{x})^2}$$

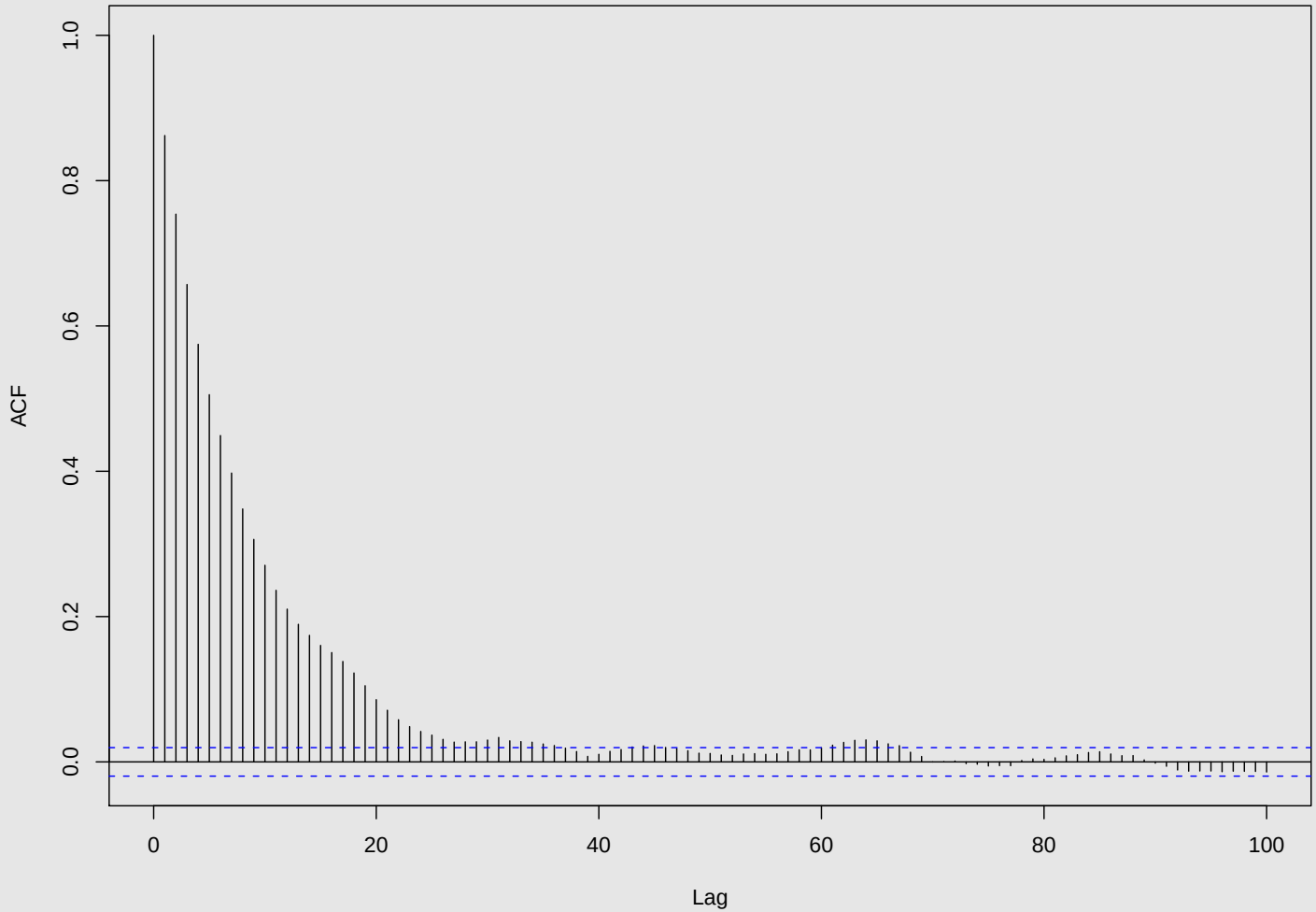
for $k = 0, 1, 2 \ldots$.

A chain with high autocorrelation for large k is typically slow to converge.

Metropolis-Hastings algorithm

Example (Gamma density)

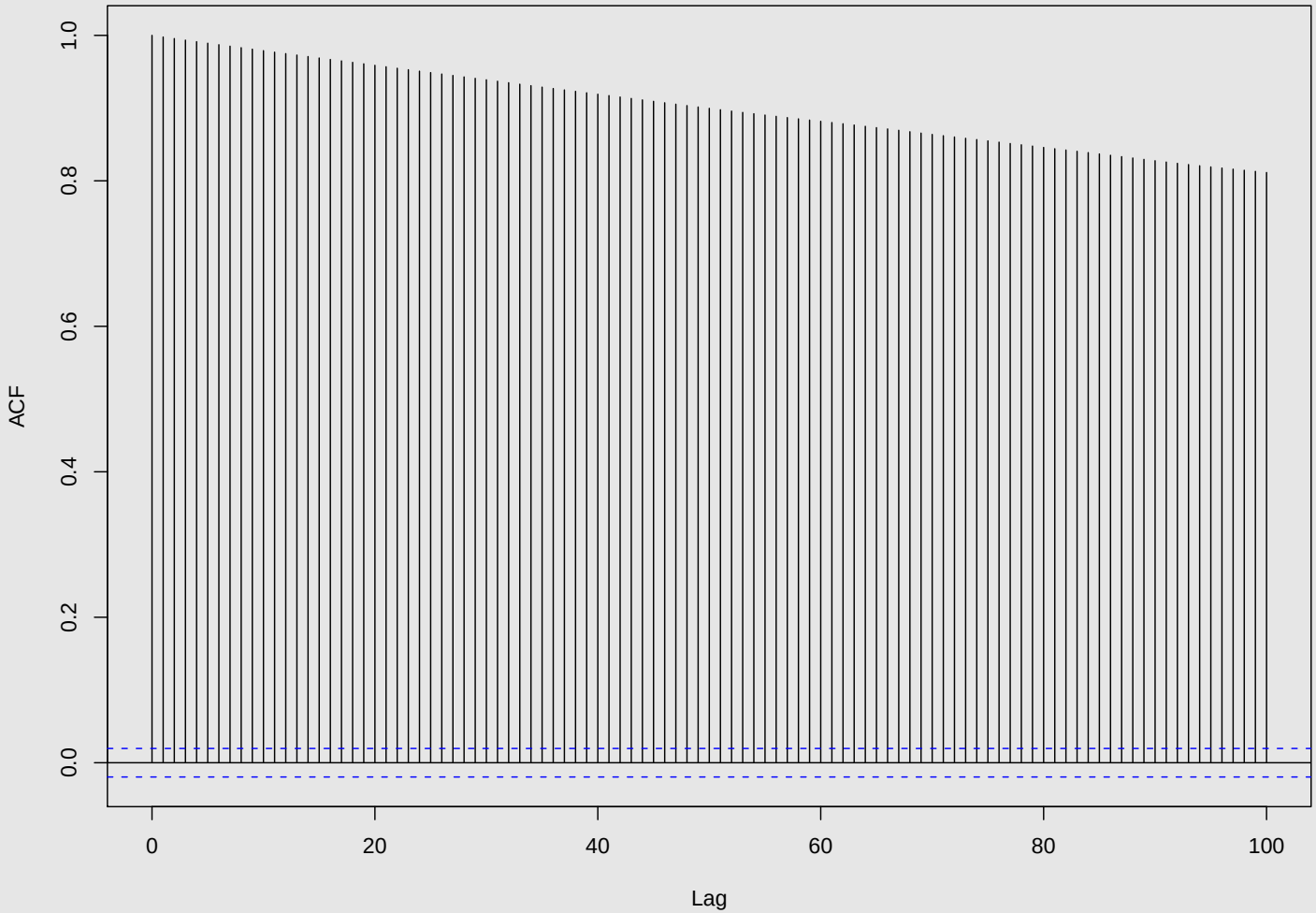
Autocorrelation function for $\sigma_q = 1$.



Metropolis-Hastings algorithm

Example (Gamma density)

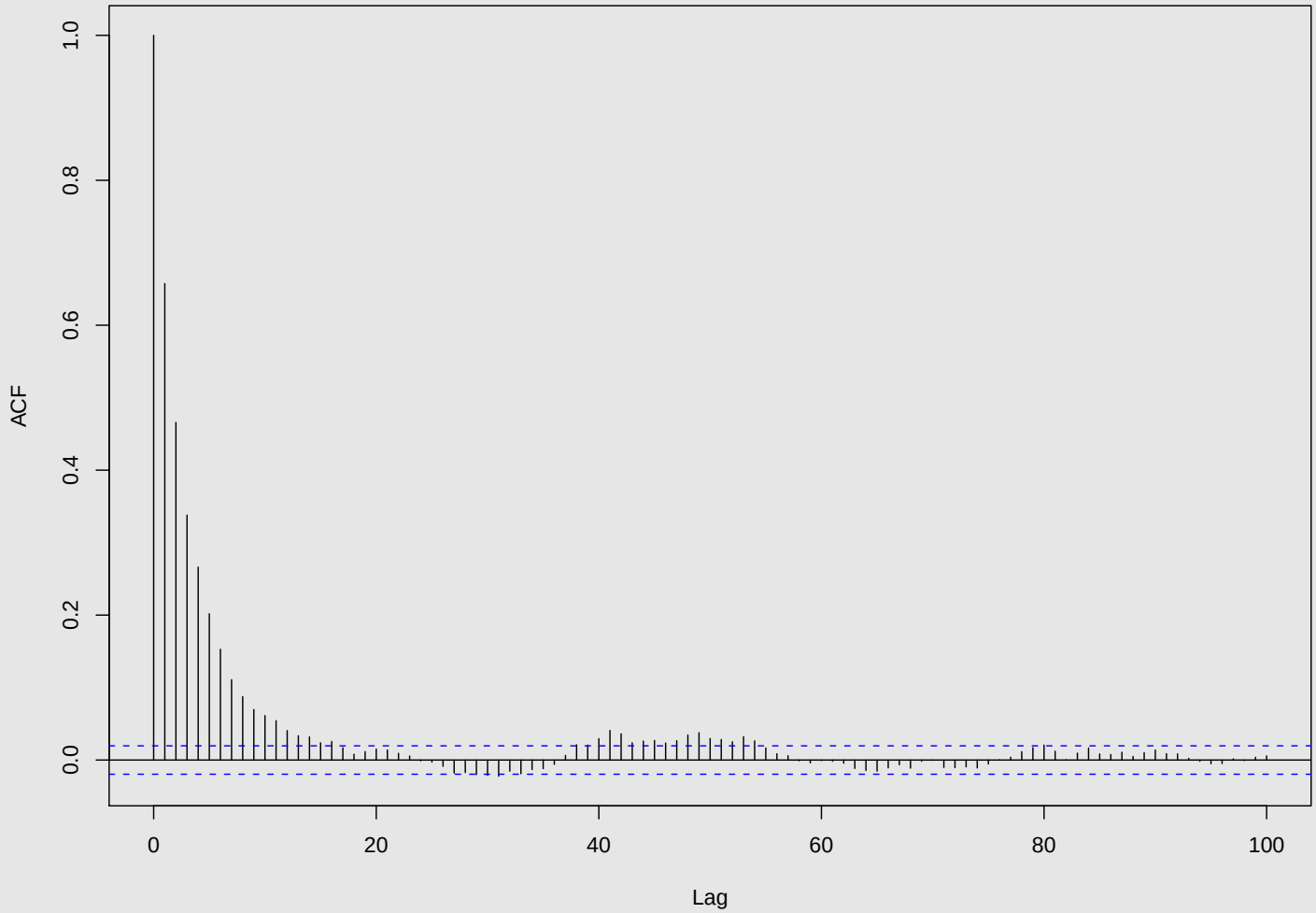
Autocorrelation function for $\sigma_q = 0.1$: inferior performance.



Metropolis-Hastings algorithm

Example (Gamma density)

Autocorrelation function for $\sigma_q = 3$: superior performance.



Metropolis-Hastings algorithm

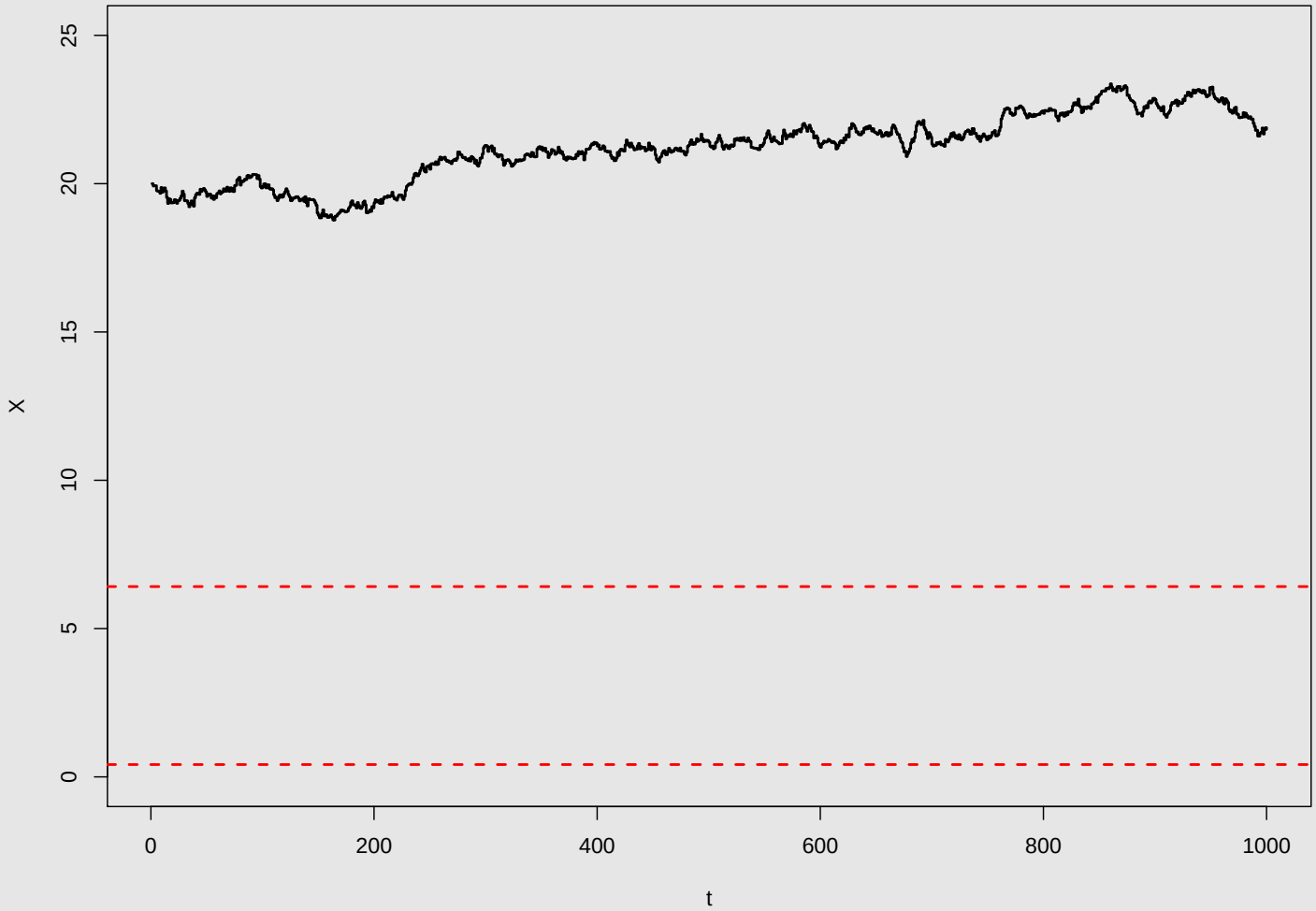
If the chain is started away from the high-probability region of π, then the it can take many steps to return there.

In the following trace plots, the red dashed lines give the 0.025 and 0.975 quantiles of the $Gamma(2.5, 1)$ distribution from the example. The chain is initialized at $X_0 = 20$, and then run for 20000 steps.

Metropolis-Hastings algorithm

Example (Gamma density)

Starting value $X_0 = 20$, $\sigma_q = 0.1$: first 1000 steps.



Metropolis-Hastings algorithm

Example (Gamma density)

Starting value $X_0 = 20$, $\sigma_q = 0.1$: first 10000 steps.



Metropolis-Hastings algorithm

Example (Gamma density)

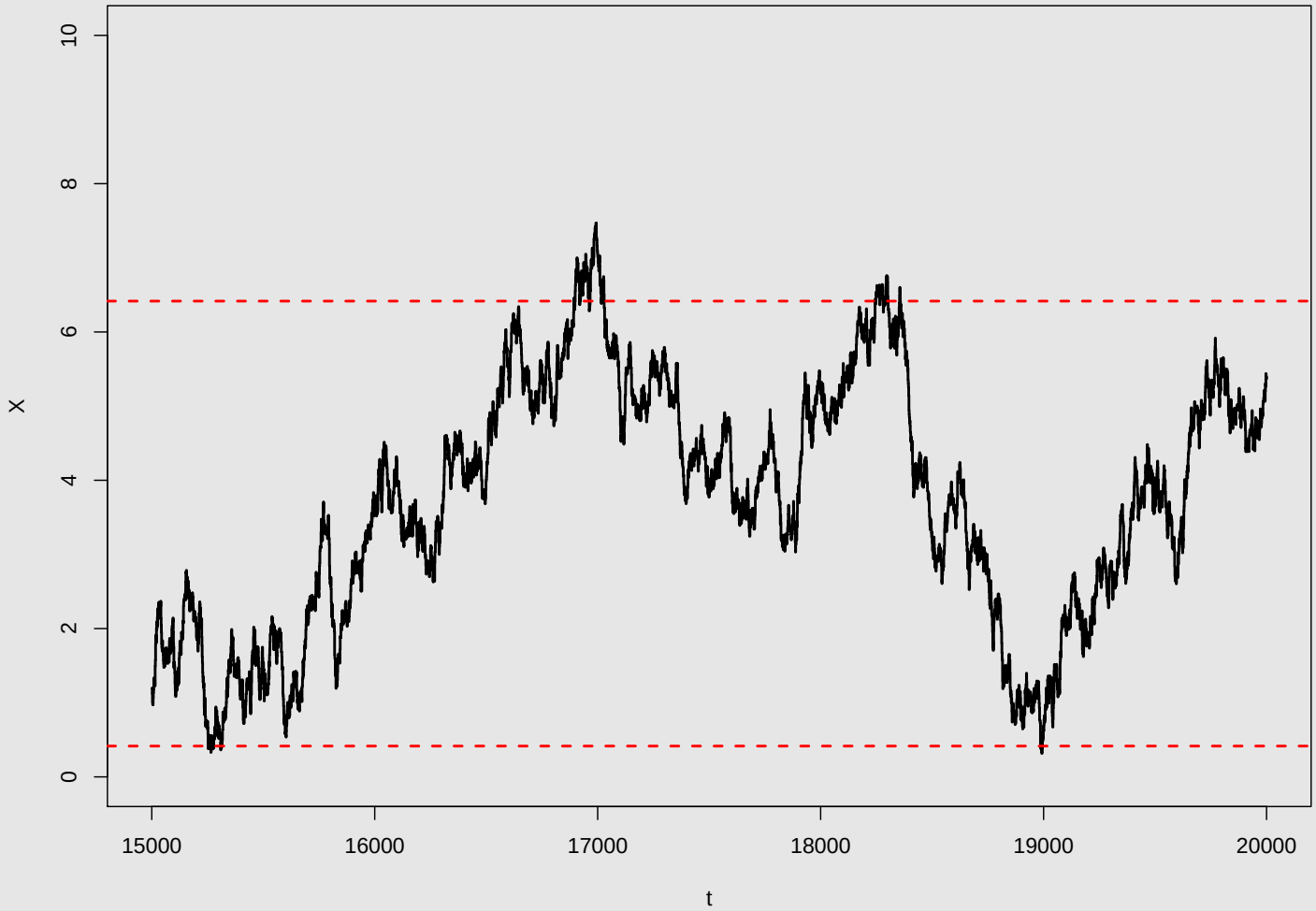
Starting value $X_0 = 20$, $\sigma_q = 0.1$: first 20000 steps.



Metropolis-Hastings algorithm

Example (Gamma density)

Starting value $X_0 = 20$, $\sigma_q = 0.1$: steps 15000 to 20000.



Metropolis-Hastings algorithm

In this final section of the chain, the generated values appear to oscillate, despite the fact that the chain has apparently reached the stationary phase. This oscillation is a result of the high autocorrelation present in the chain.

It is often difficult to distinguish such high autocorrelation from the case where a chain has not converged.

The high autocorrelation results here from the choice $\sigma_q = 0.1$; this value is smaller than is optimal.

Gibbs Sampler

The Metropolis-Hastings algorithm above is valid for both univariate and multivariate probability distributions, but is more complicated in high dimensions.

The objective is to choose a transition density q that moves around the space $\mathbb{X}$ quickly, which means that we wish to have the acceptance probability reasonably large.

In high dimensions, this is often difficult to achieve. The *Gibbs Sampler* algorithm attempts to solve this problem by breaking down a high-dimensional problem into several lower dimensional problems that are solved iteratively and simultaneously.

Gibbs Sampler

Suppose that π is a probability density in K dimensions, and let the variables be denoted $(X_1, ..., X_K)$. Define the conditional density $\pi_k(.|.)$ for

$$X_k|X_1, ..., X_{k-1}, X_{k+1}, ...X_K$$

by

$$\pi_k\left(x_k; x_{(k)}\right) = \frac{\pi\left(x_1, ..., x_K\right)}{\pi\left(x_1, ..., x_{k-1}, x_{k+1}, ...x_K\right)} \propto \pi\left(x_1, ..., x_K\right)$$

where the denominator is the marginal distribution of $X_{(k)}$, the $K-1$ variables excluding X_k.

The Gibbs Sampler utilizes this set of K *full conditional distributions* to construct a Markov chain on the support of the joint distribution.

Gibbs Sampler

It is implemented using the following algorithm:

1. Set a vector of starting values for the K variables $(x_{10}, ..., x_{K0})$.
2. Sample in turn from the conditional distributions as follows:

   (a) sample x_{11} from $\pi_1 (x_1; x_{20}, x_{30}, ..., x_{K0})$
   (b) sample x_{21} from $\pi_2 (x_2; x_{11}, x_{30}, ..., x_{K0})$
   (c) sample x_{31} from $\pi_3 (x_3; x_{11}, x_{21}, ..., x_{K0})$

   $\vdots$

   (K) sample x_{K1} from $\pi_K (x_K; x_{11}, x_{21}, ..., x_{K-1\ 1})$

   This completes one step of the Gibbs sampler.
3. Return to 2 (a), and repeat to obtain, at step t, the sampled variates $(x_{1t}, x_{2t}, ..., x_{Kt})$

Gibbs Sampler

This Markov chain defines, in steps 2(a)-2(K), a means of updating the vector x_t to vector x_{t+1}. Each of the steps can be achieved using direct sampling from the conditional distribution if that is possible, but can also involve individual Metropolis-Hastings steps, with acceptance probabilities

$$\alpha_k(x, z) = \min\left\{1, \frac{\pi_k\left(z; x_{(k)}\right)}{\pi_k\left(x; x_{(k)}\right)} \frac{q_k(z, x)}{q_k(x, z)}\right\}$$

for $k = 1, ..., K$. In 2., the steps can also be completed in *random order*

Finally, these steps can be achieved with the scalar variables X_1,...,X_K or with these components as vector variables; deciding on which *blocks* of variables to update simultaneously is often a key issue.

Gibbs Sampler as Metropolis-Hastings

The Gibbs sampler is in fact a special case of the Metropolis-Hastings algorithm: we can regard the individual updates in 2. as implementing K separate transition kernels $P_1, \ldots, P_K$ that act on the components of the vector $\boldsymbol{X}$; note that these kernels in isolation yield *reducible* Markov chains.

A more general form of Metropolis-Hastings algorithm is based on a *mixture* transition kernel

$$P(x, B) = \sum_j \omega_j P_j(x, B)$$

where $0 \leqslant \omega_j \leqslant 1$ and $\sum_j \omega_j = 1$, and the P_j are themselves transition kernels. This allows for the possibility of choosing several proposal densities q_j.

Gibbs Sampler: example

Example (Bivariate Normal)

Suppose $\boldsymbol{X} = (X_1, X_2)^\intercal \sim \mathcal{N}_2(\mathbf{0}, \Sigma)$ where

$$\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

The general result for multivariate normal distribution conditional distributions is that if $\boldsymbol{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2)^\intercal \sim \mathcal{N}_d(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{X}_1$ is $(d_1 \times 1)$, and

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$$

then

$$\boldsymbol{X}_1 | \boldsymbol{X}_2 = \boldsymbol{x}_2 \mathcal{N}_{d_1} \sim \left(\boldsymbol{\mu}_1 + \Sigma_{12}\Sigma_{22}^{-1}(\boldsymbol{x}_2 - \boldsymbol{\mu}_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \right)$$

Gibbs Sampler: example

Example (Bivariate Normal)

Therefore, here $d = 2$, and

$$\begin{aligned} X_1|X_2 = x_2 &\sim \mathcal{N}(\rho x_2, (1-\rho^2)) \\ X_2|X_1 = x_1 &\sim \mathcal{N}(\rho x_1, (1-\rho^2)) \end{aligned}$$

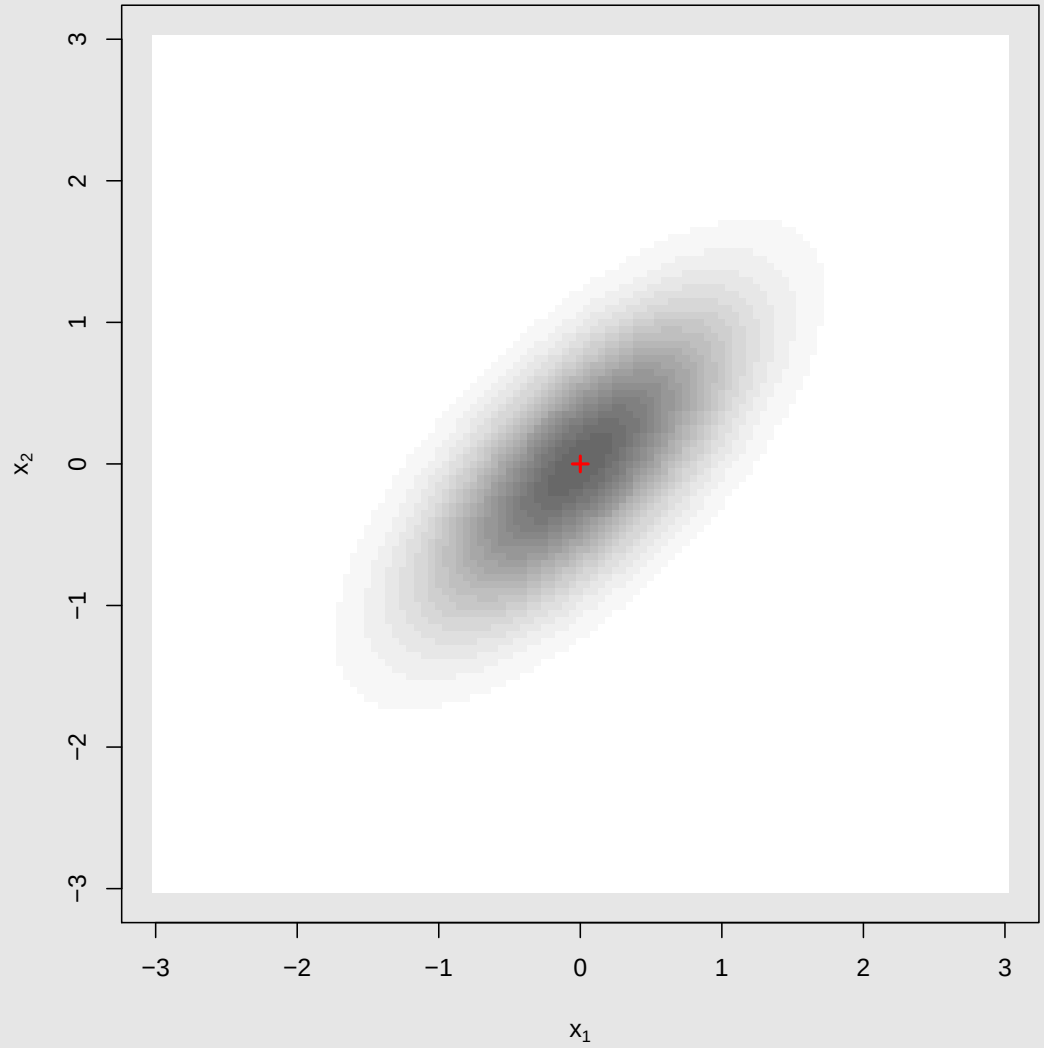
These distributions are sampled repeatedly with updating of the conditioning value after each sampling.

Suppose we start at $(x_{10}, x_{20}) = (0, 0)$.

Gibbs Sampler: example

Example (Bivariate Normal)

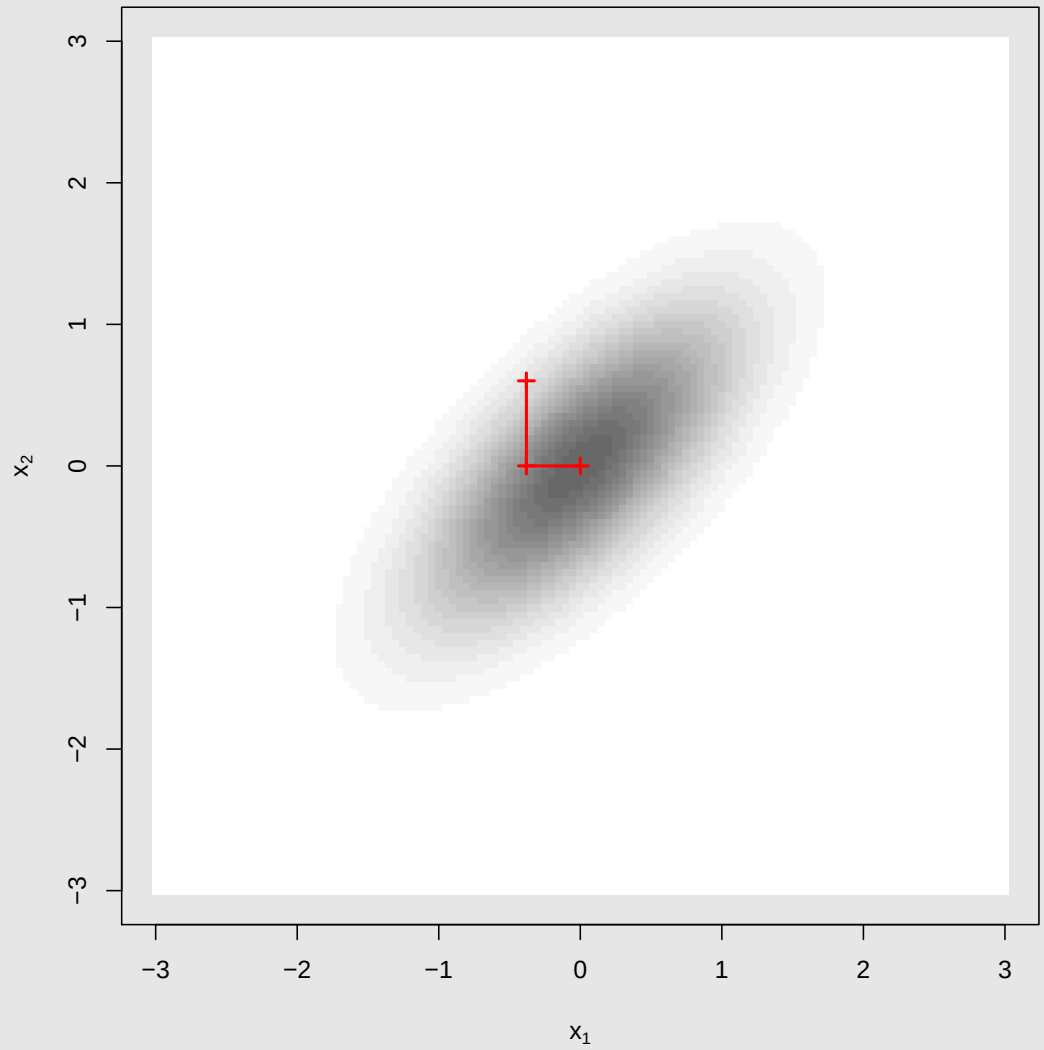
Initial point: (x_{10}, x_{20})



Gibbs Sampler: example

Example (Bivariate Normal)

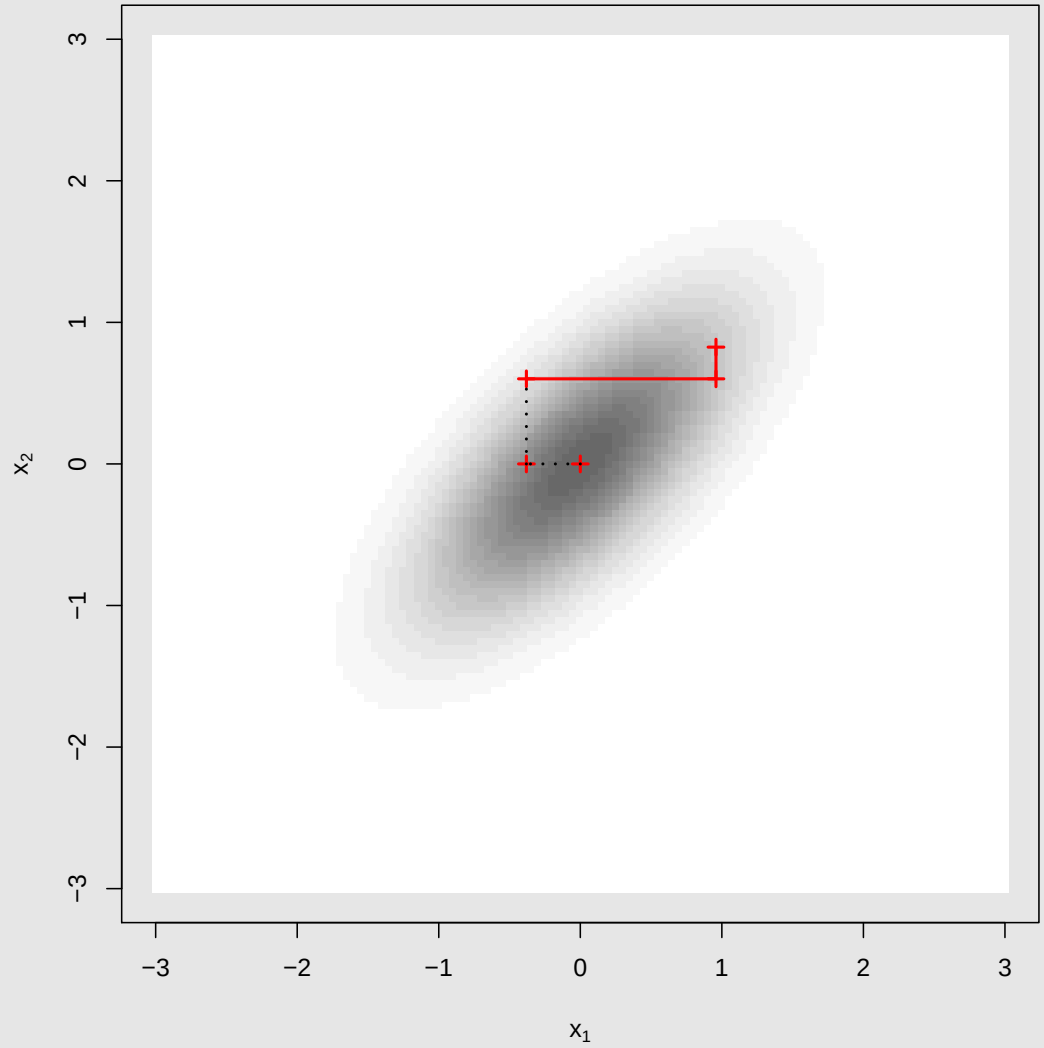
After one update: (x_{11}, x_{21})



Gibbs Sampler: example

Example (Bivariate Normal)

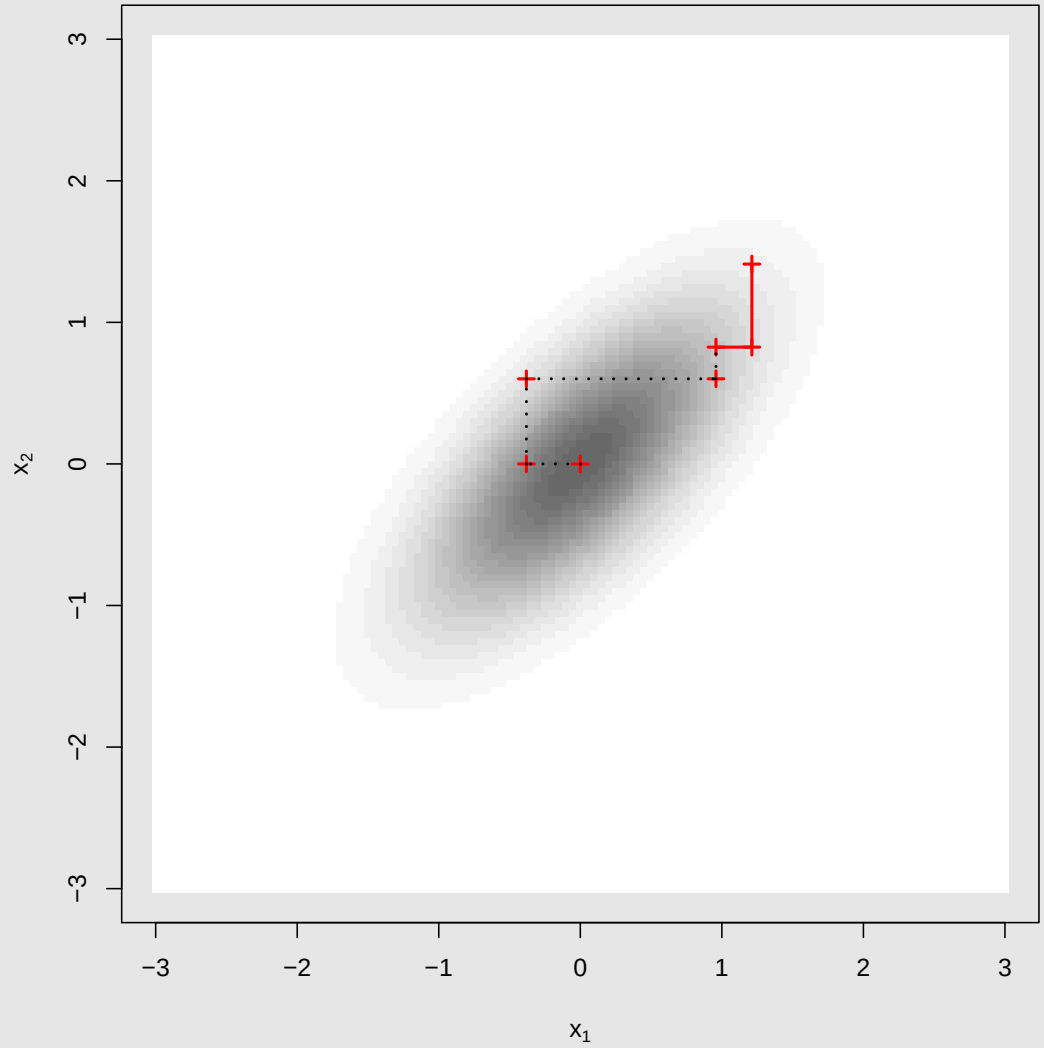
After two updates: (x_{12}, x_{22})



Gibbs Sampler: example

Example (Bivariate Normal)

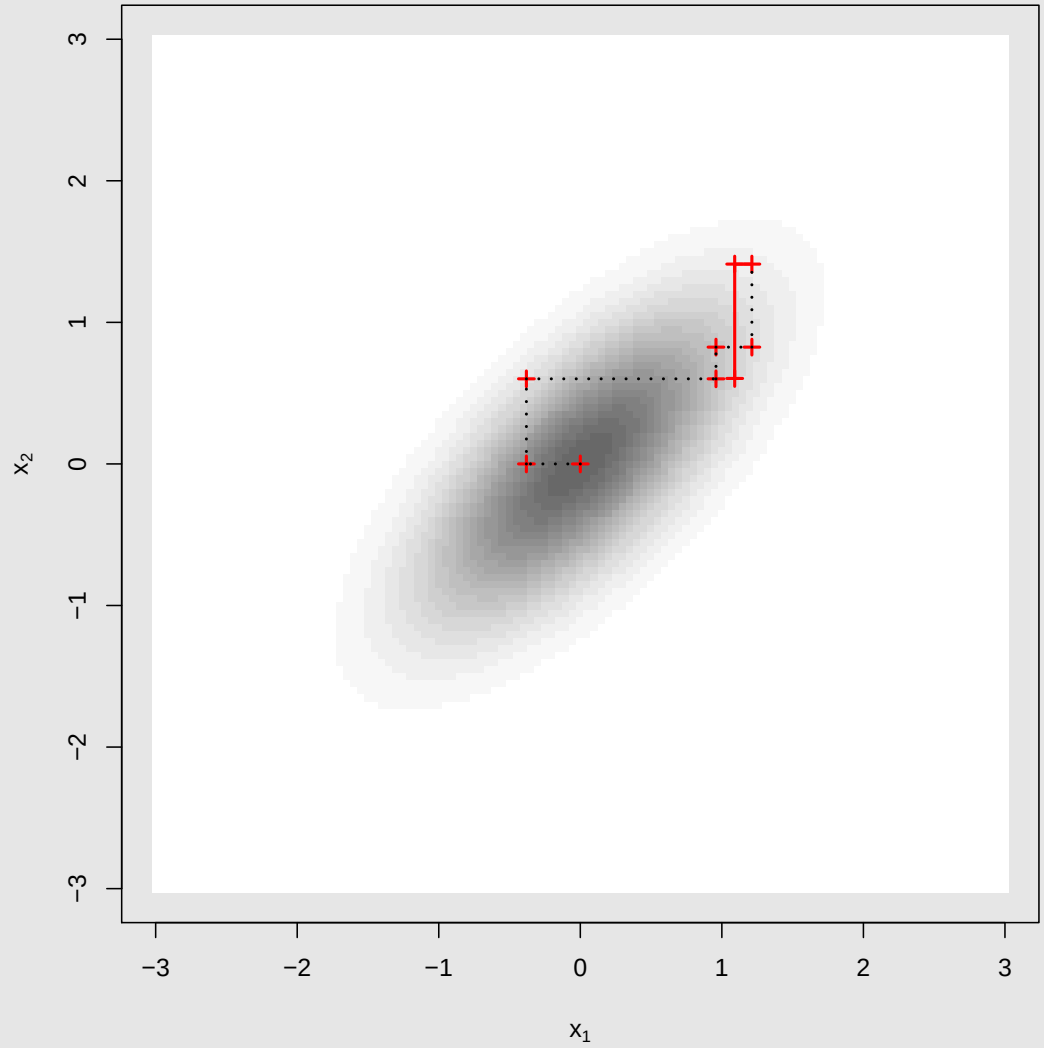
After three updates: (x_{13}, x_{23})



Gibbs Sampler: example

Example (Bivariate Normal)

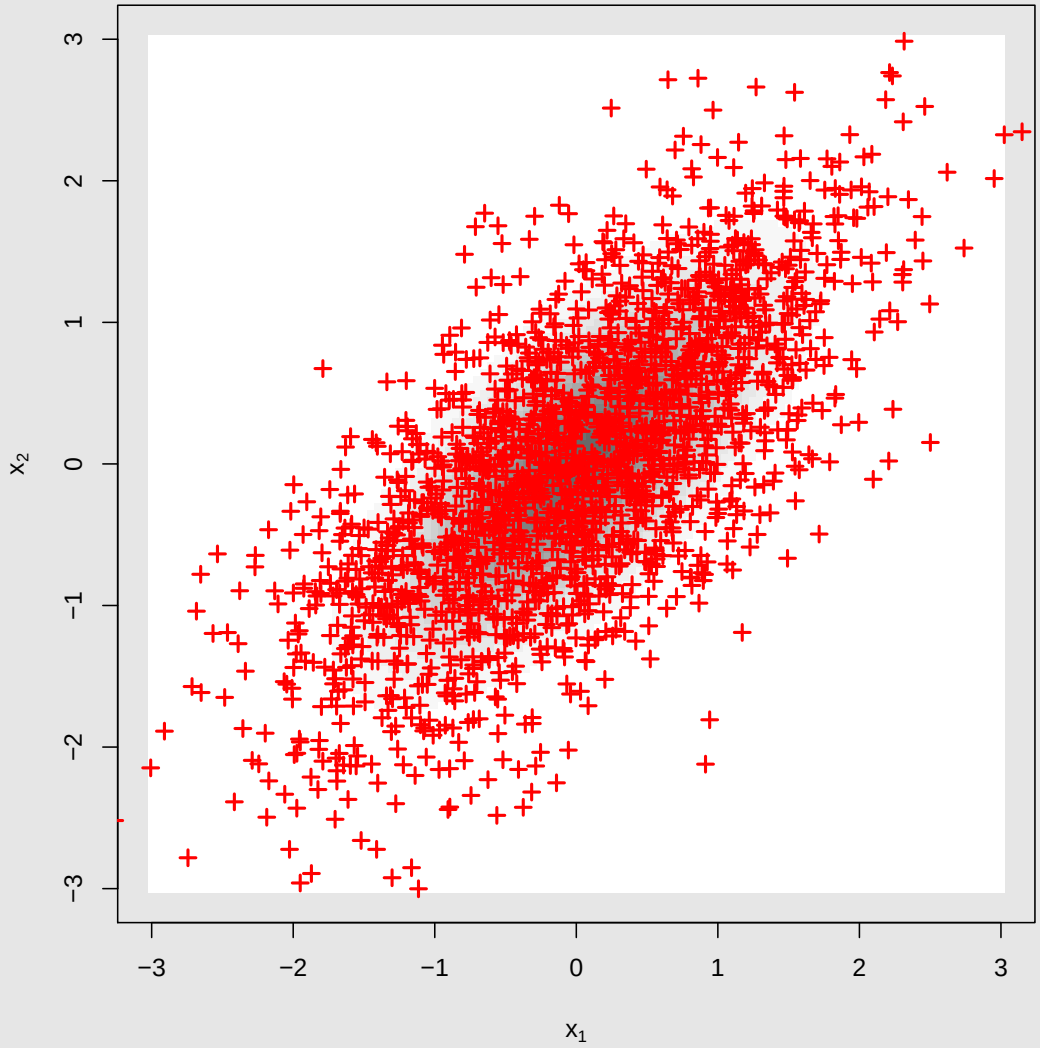
After four updates: (x_{14}, x_{24})



Gibbs Sampler: example

Example (Bivariate Normal)

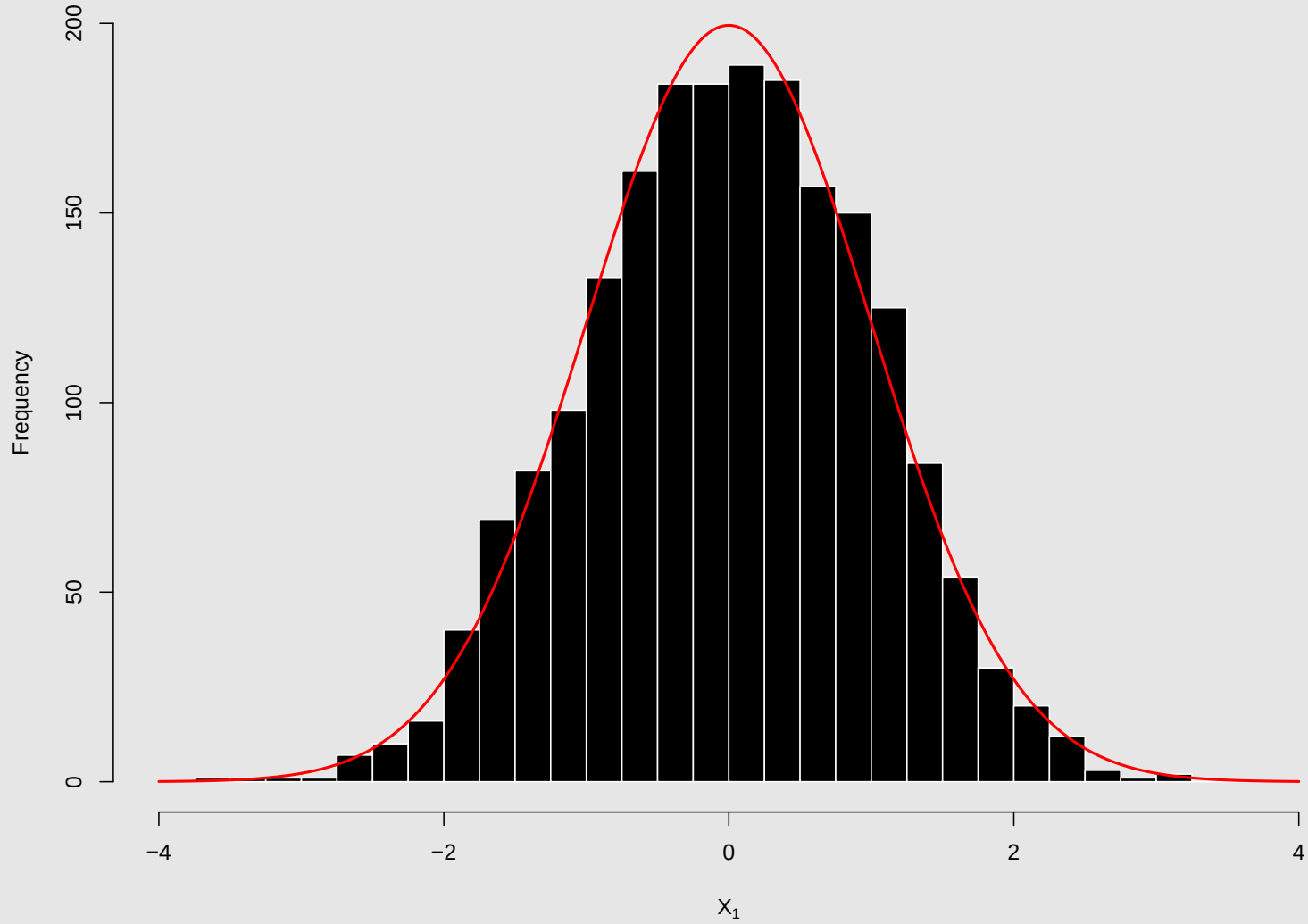
After 2000 updates: entire collected sample



Gibbs Sampler: example

Example (Bivariate Normal)

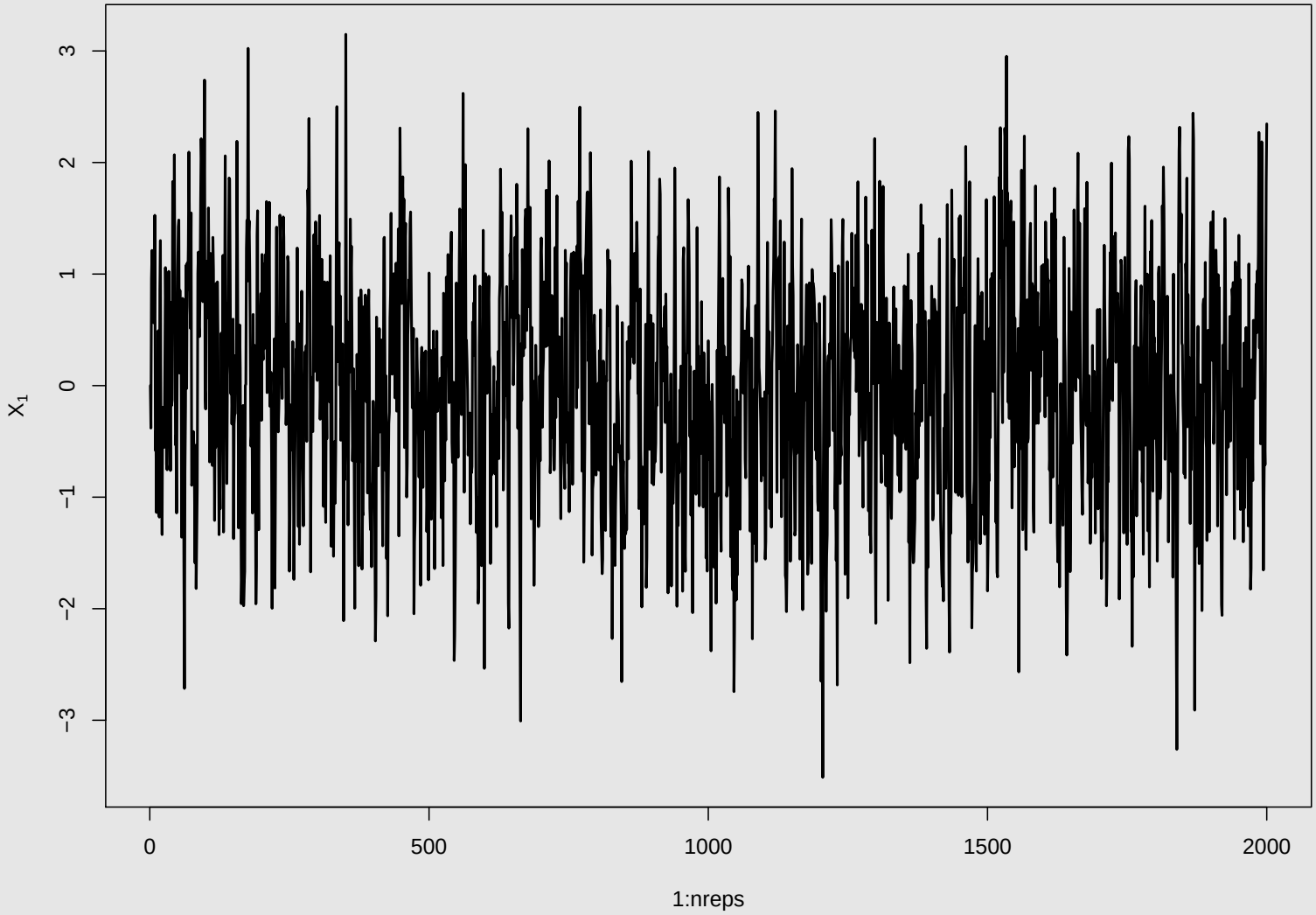
Histogram for X_1 with true marginal density (solid):



Gibbs Sampler: example

Example (Bivariate Normal)

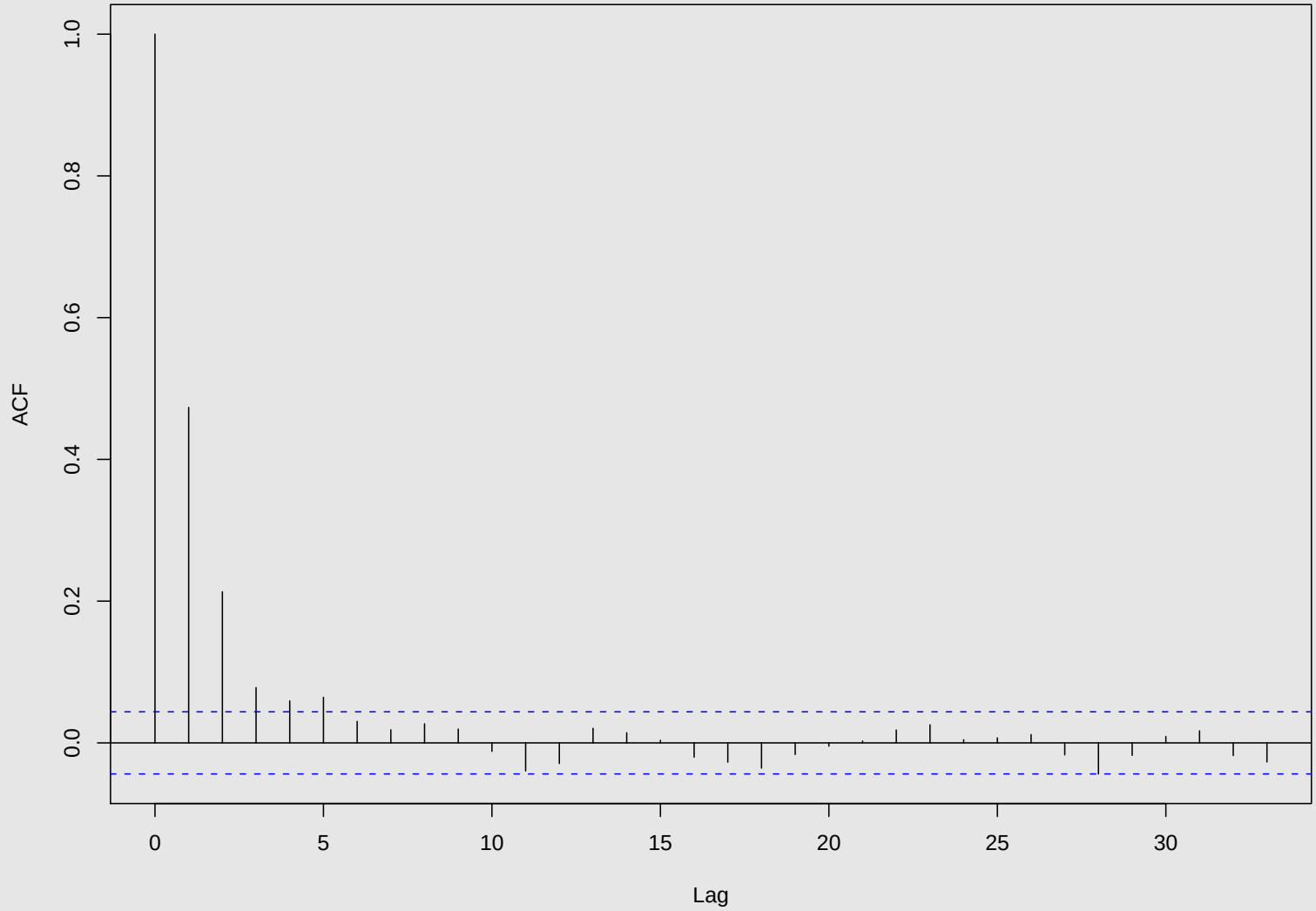
Trace for X_1:



Gibbs Sampler: example

Example (Bivariate Normal)

ACF for X_1:



Gibbs Sampler: example

Example (Bivariate Normal)

We note that

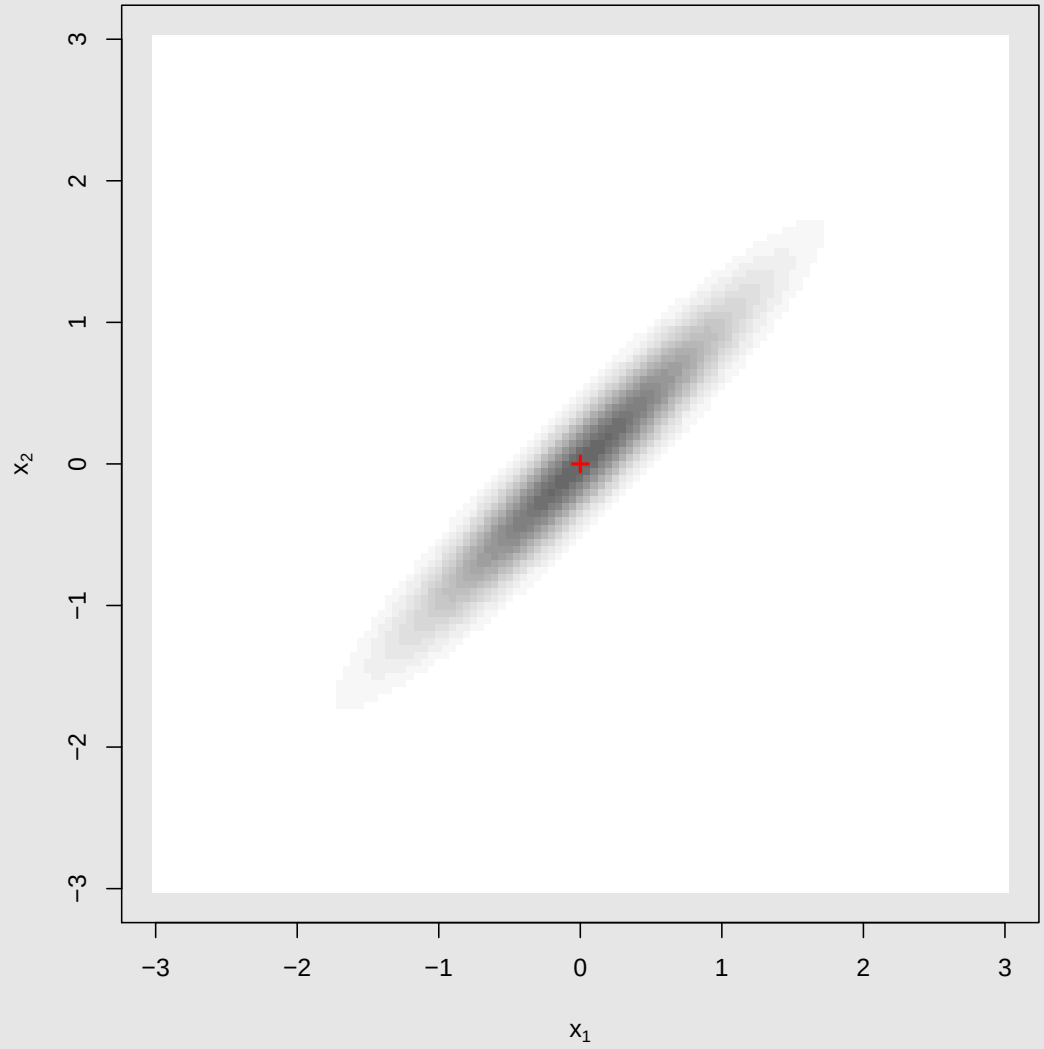
- the Gibbs sampler makes one-step moves along the coordinate axes
- the moves can traverse the support of the joint density fairly well
- there is no "tuning" of a proposal parameter (like σ_q)
- the samples of x_1 that are collected across steps are (dependent) samples from the correct marginal distribution for X_1; the same result holds for X_2

We can re-run the Gibbs sampler from the same starting value, but now with $\rho = 0.95$.

Gibbs Sampler: example

Example (Bivariate Normal)

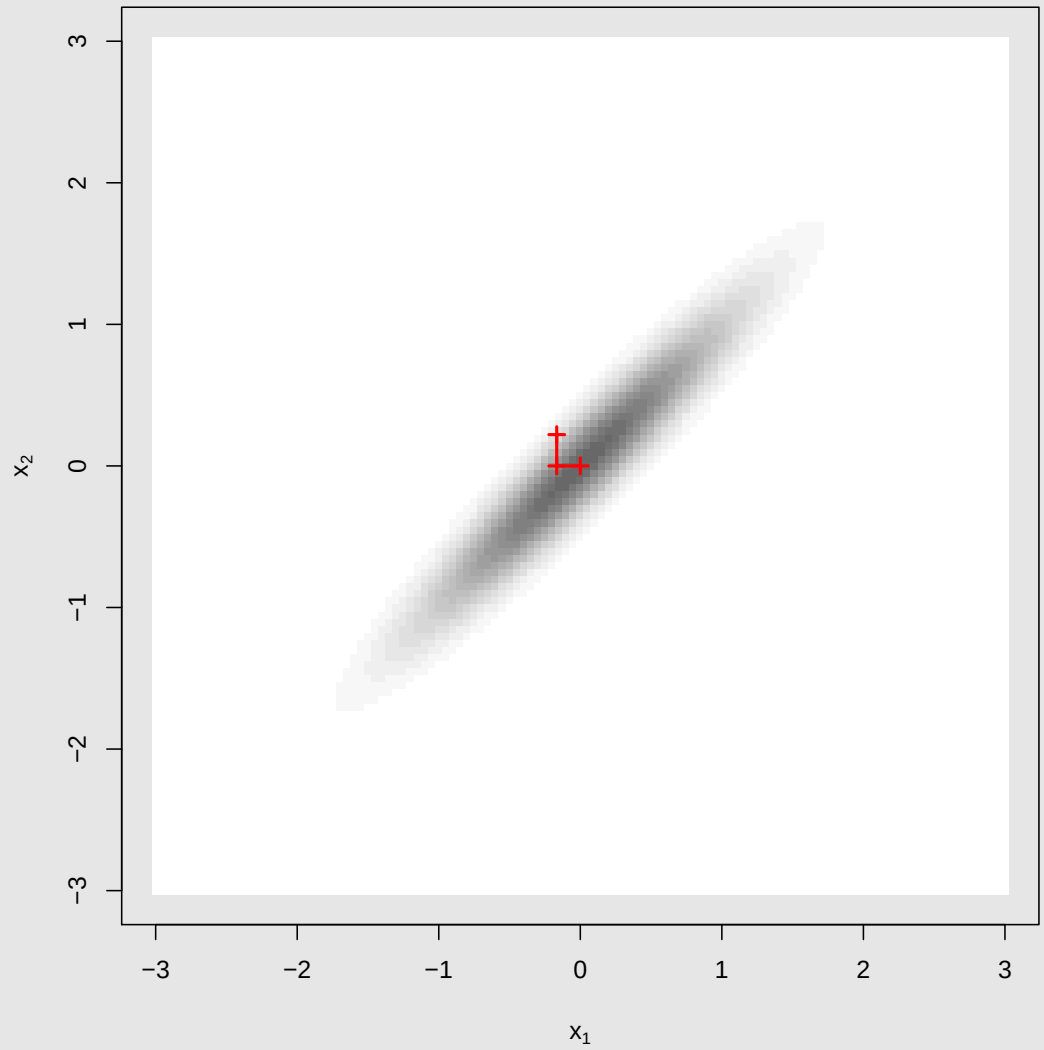
Initial point: (x_{10}, x_{20})



Gibbs Sampler: example

Example (Bivariate Normal)

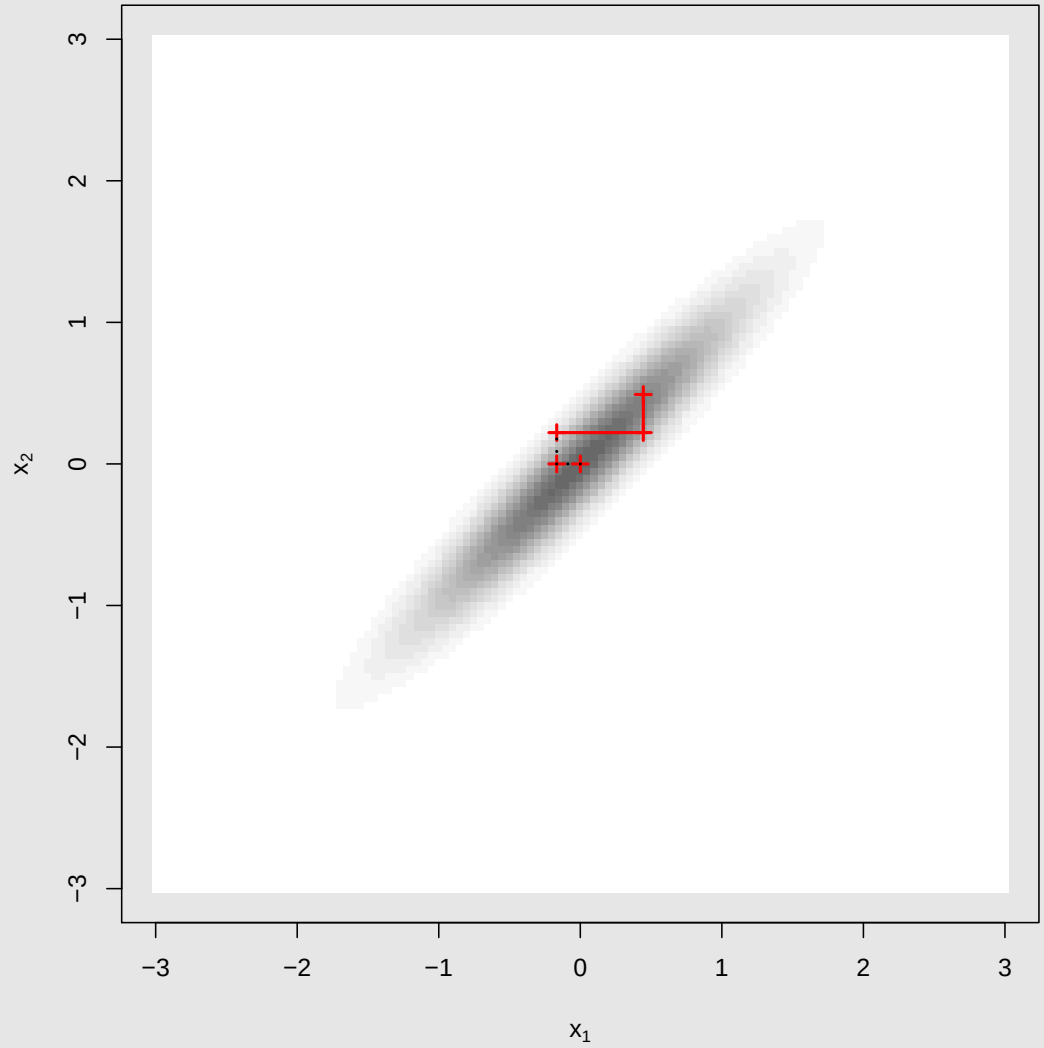
After one update: (x_{11}, x_{21})



Gibbs Sampler: example

Example (Bivariate Normal)

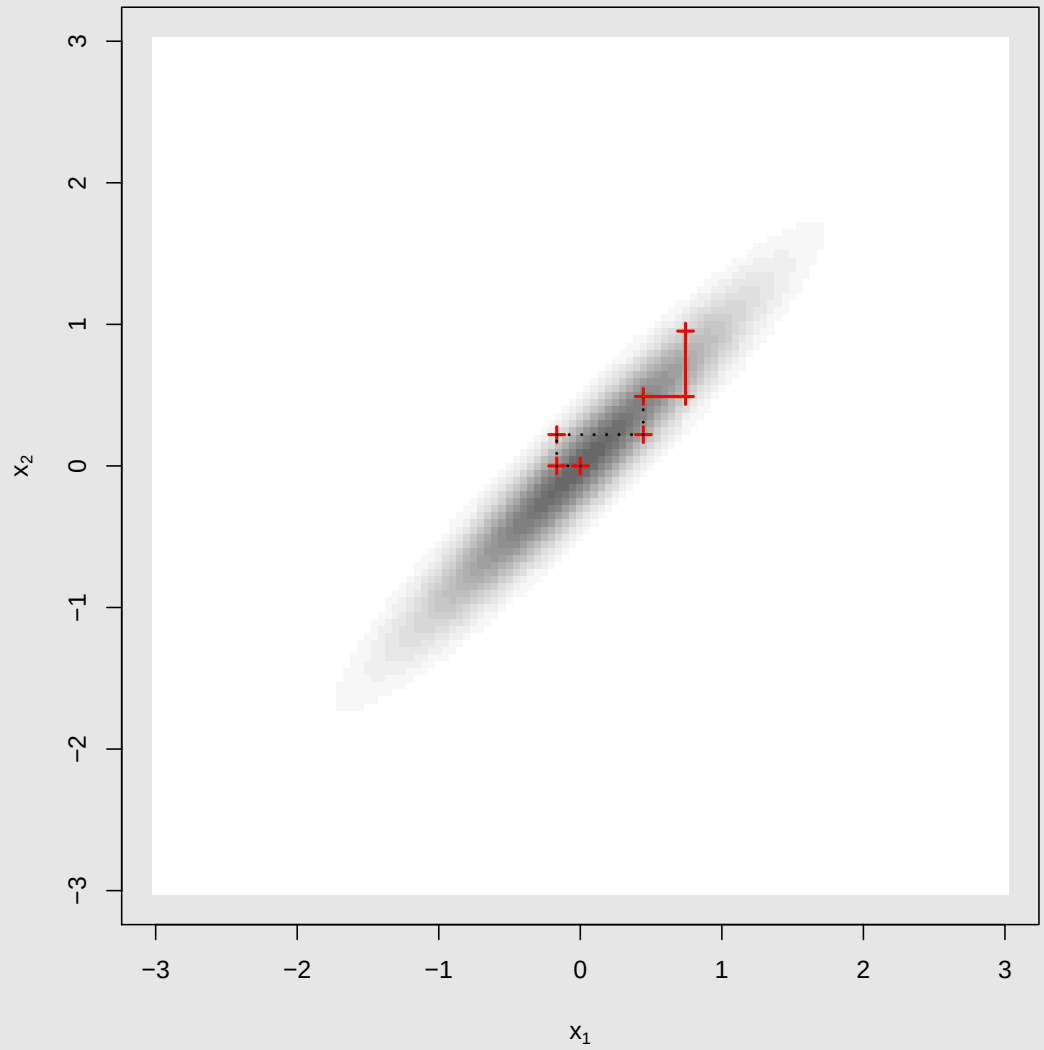
After two updates: (x_{12}, x_{22})



Gibbs Sampler: example

Example (Bivariate Normal)

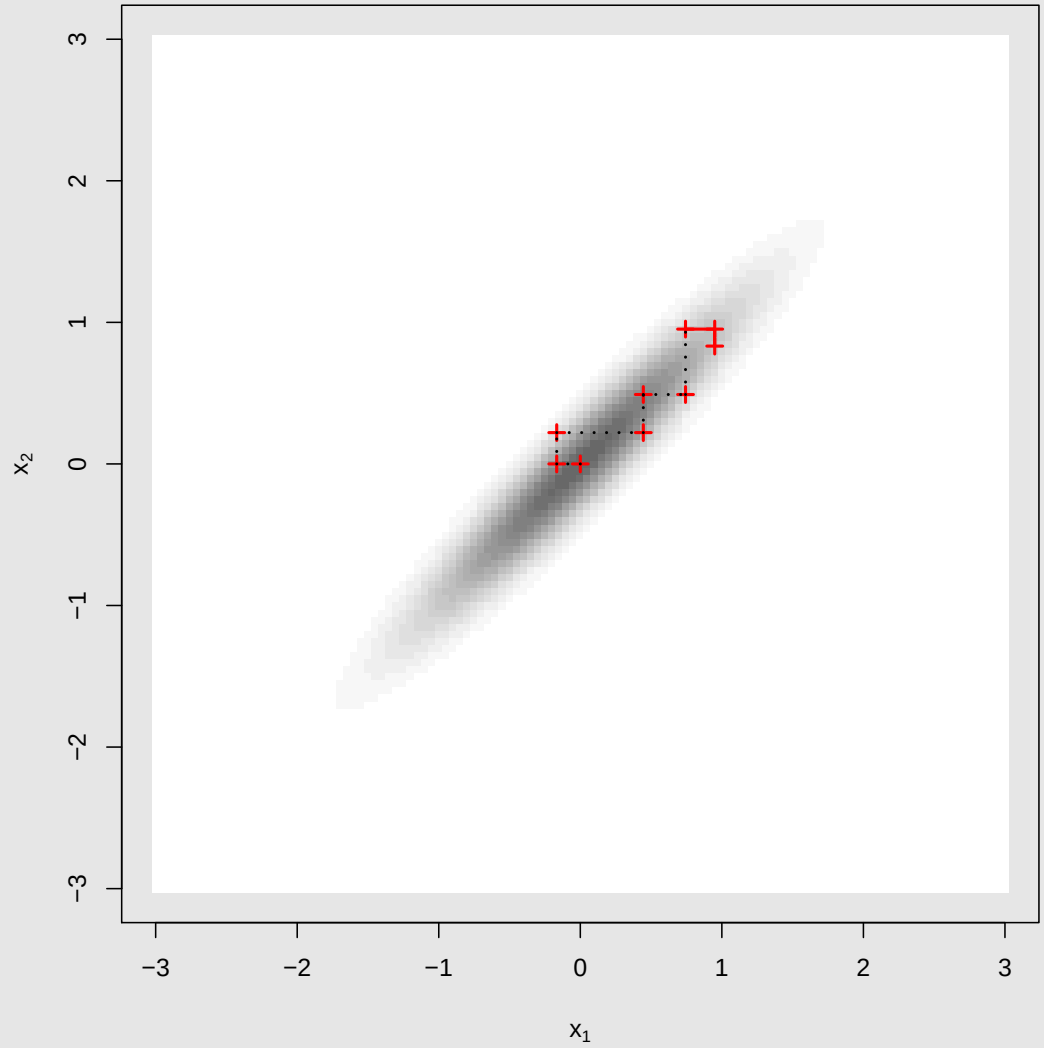
After three updates: (x_{13}, x_{23})



Gibbs Sampler: example

Example (Bivariate Normal)

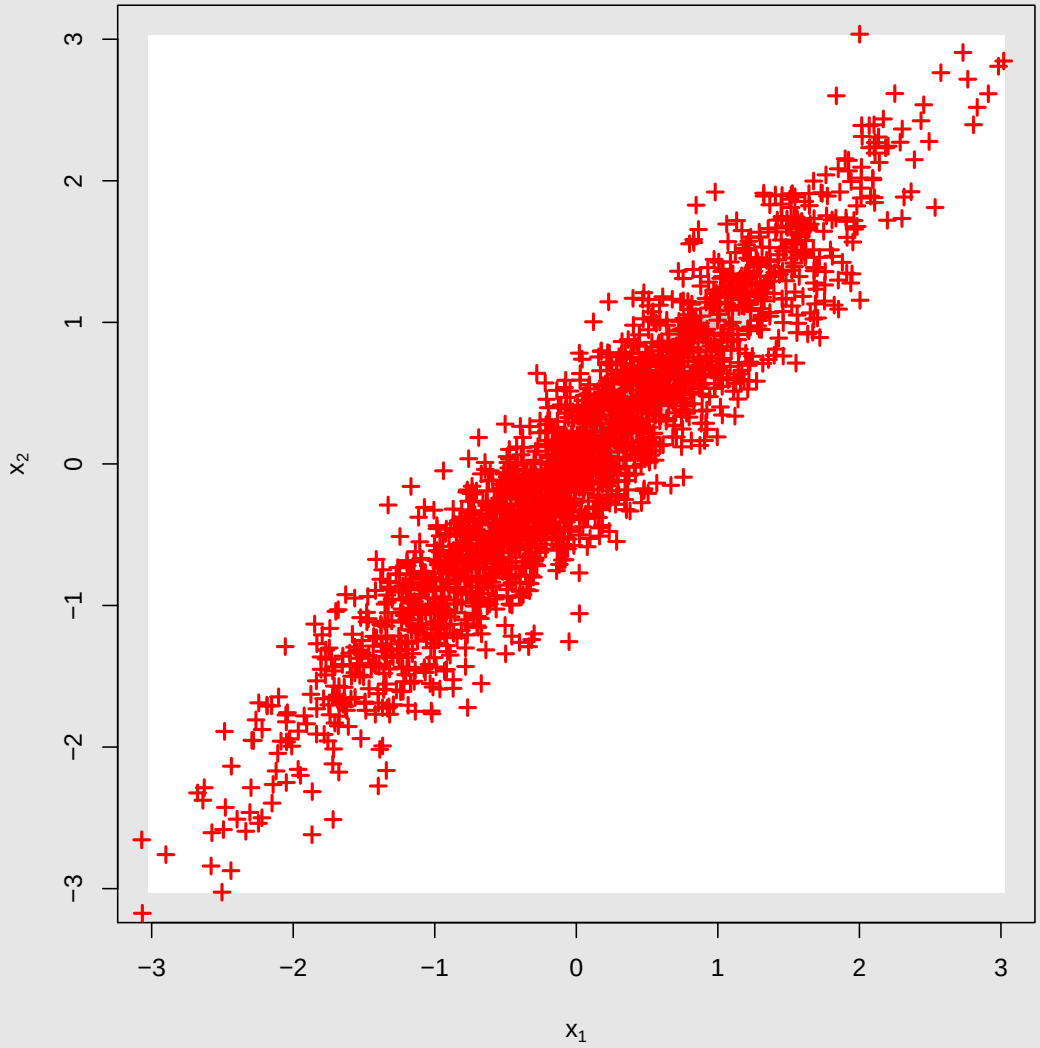
After four updates: (x_{14}, x_{24})



Gibbs Sampler: example

Example (Bivariate Normal)

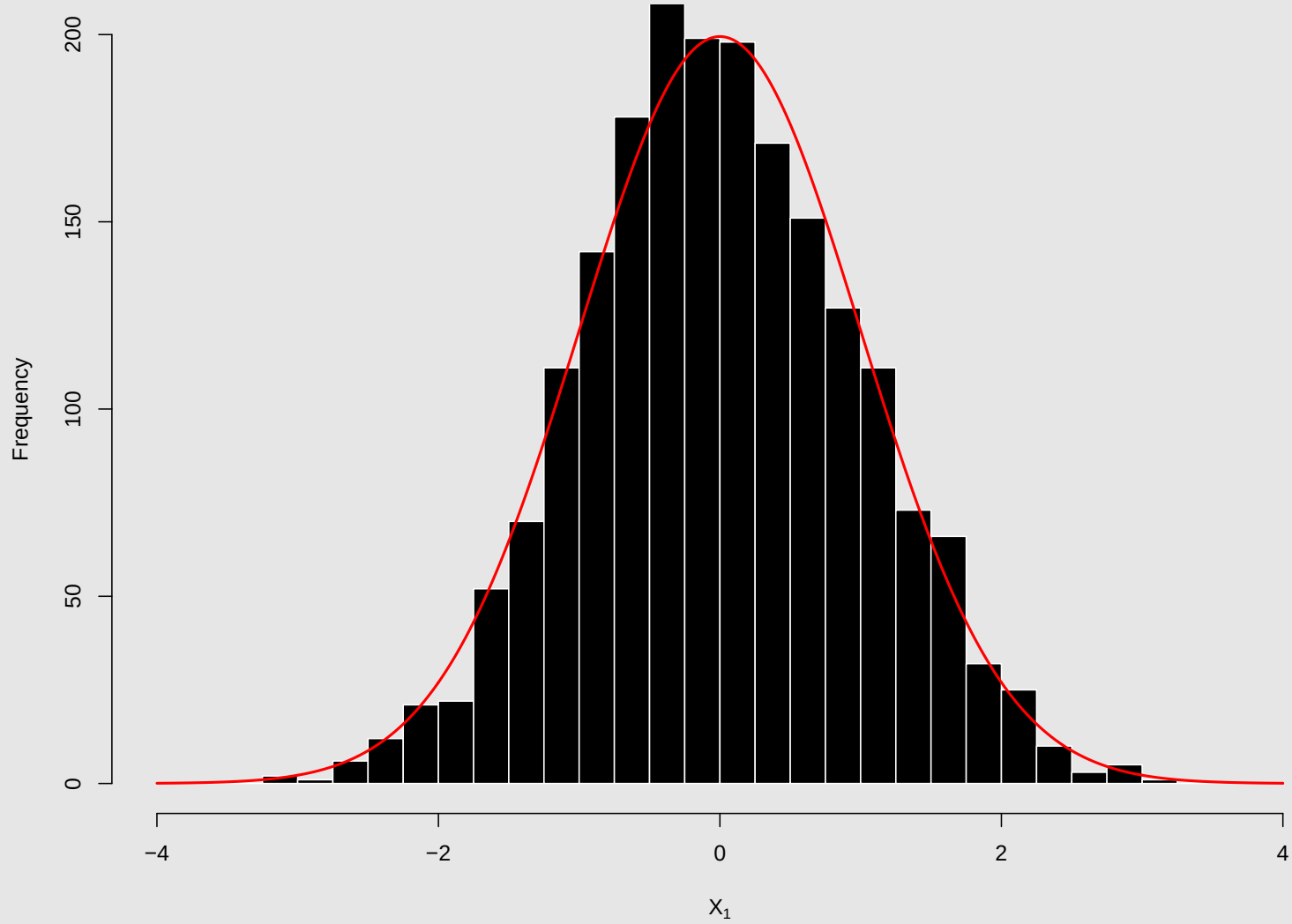
After 2000 updates: entire collected sample



Gibbs Sampler: example

Example (Bivariate Normal)

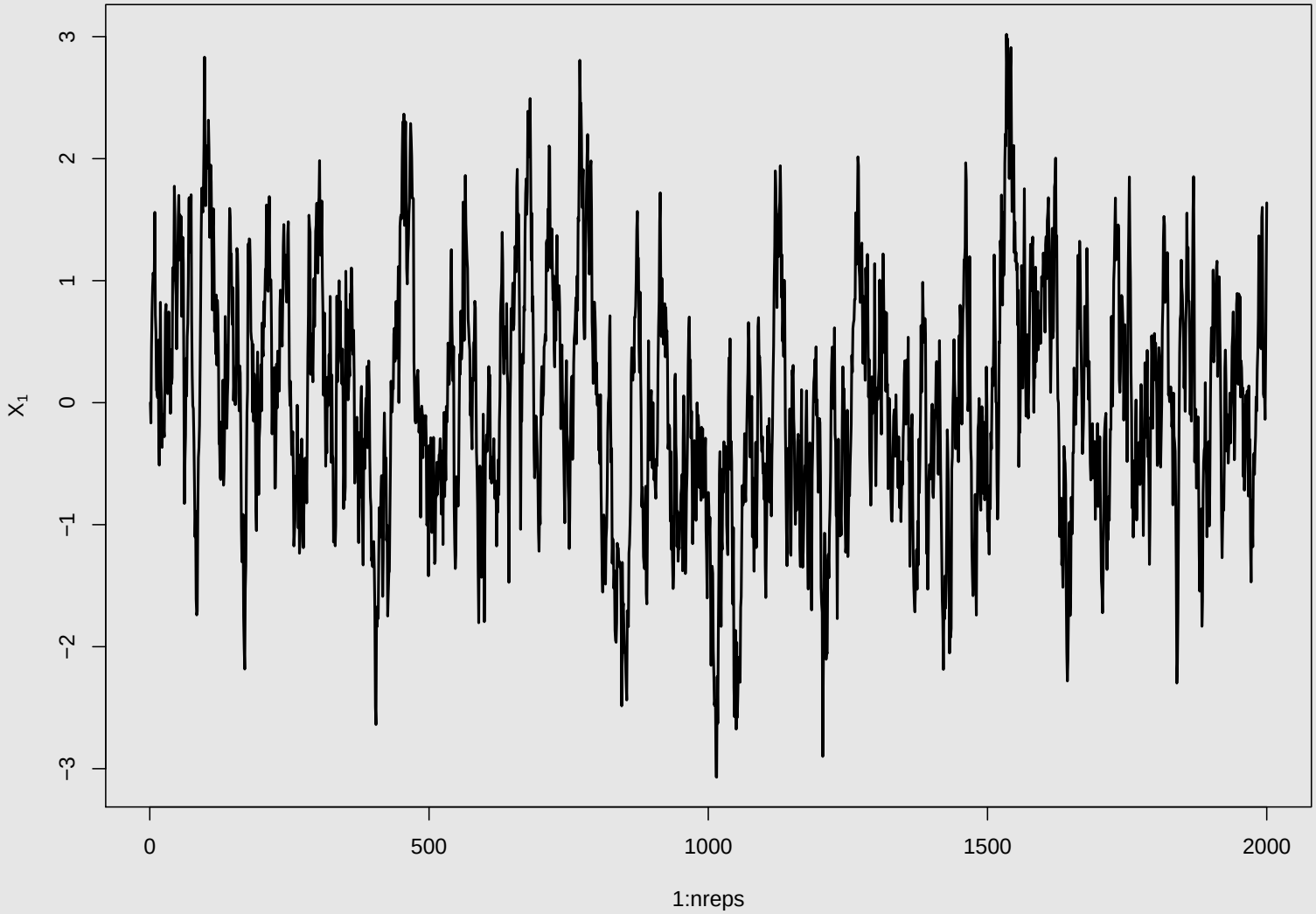
Histogram for X_1 with true marginal density (solid):



Gibbs Sampler: example

Example (Bivariate Normal)

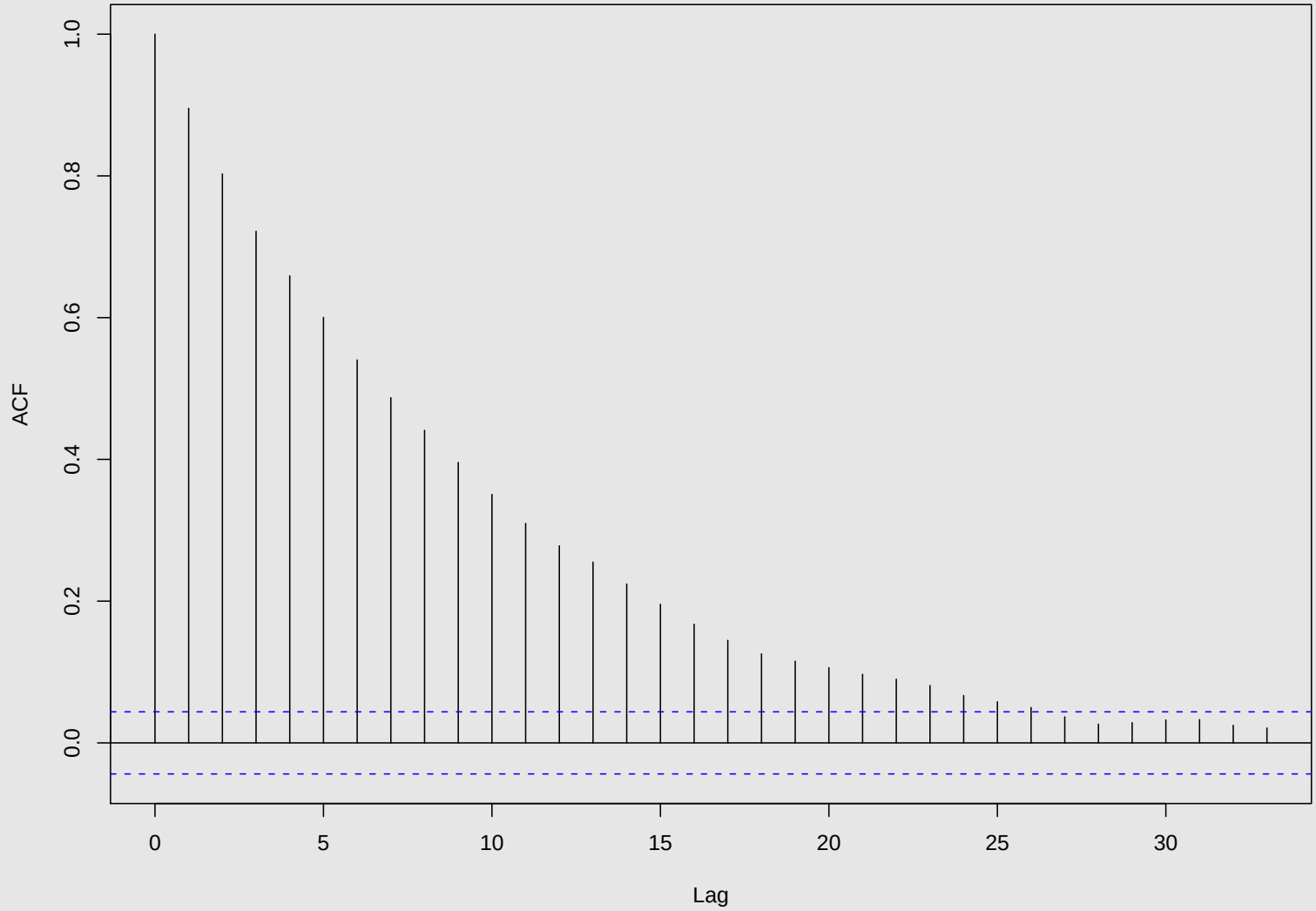
Trace for X_1:



Gibbs Sampler: example

Example (Bivariate Normal)

ACF for X_1:



Gibbs Sampler: example

Example (Bivariate Normal)

With $\rho = 0.95$, the Gibbs sampler still performs adequately, but the moves made are smaller, and exploring the distribution is much more difficult.

This illustrates a potential general problem with the Gibbs sampler: although it is straightforward to implement as it involves only sampling variates from univariate densities, the restriction to moves along the coordinate axes can cause problems if the variables are **highly correlated**.

Gibbs Sampler: example

Example (Weibull posterior distribution)

Suppose that $Y_1, \ldots, Y_n$ are i.i.d. from the Weibull distribution with density

$$f_Y(y; \gamma_0, \lambda_0) = \gamma_0 \lambda_0 y^{\gamma_0 - 1} \exp\{-\lambda_0 y^{\alpha_0}\} \qquad y > 0$$

and zero otherwise, for parameters $\gamma_0, \lambda_0 > 0$. We seek to perform Bayesian inference for the two unknown parameters.

The likelihood function for observed data $y_1, \ldots, y_n$ takes the form

$$L(\gamma_0, \lambda_0; \boldsymbol{y}) = \gamma_0^n \lambda_0^n \left(\prod_{i=1}^n y_i \right)^{\gamma_0 - 1} \exp\left\{ -\lambda_0 \sum_{i=1}^n y_i^{\alpha_0} \right\}$$

Gibbs Sampler: example

Example (Weibull posterior distribution)

We assume independent $Exponential(0.1)$ priors for γ_0 and λ_0

$$p(\gamma_0, \lambda_0) = 0.01e^{-0.1(\gamma_0+\lambda_0)} \qquad \gamma_0, \lambda_0 > 0.$$

This yields the posterior distribution up to proportionality as

$$\pi(\gamma_0, \lambda_0|\boldsymbol{y}) \propto L(\gamma_0, \lambda_0; \boldsymbol{y}) p(\gamma_0, \lambda_0) \qquad \gamma_0, \lambda_0 > 0$$

which is a non-standard distribution.

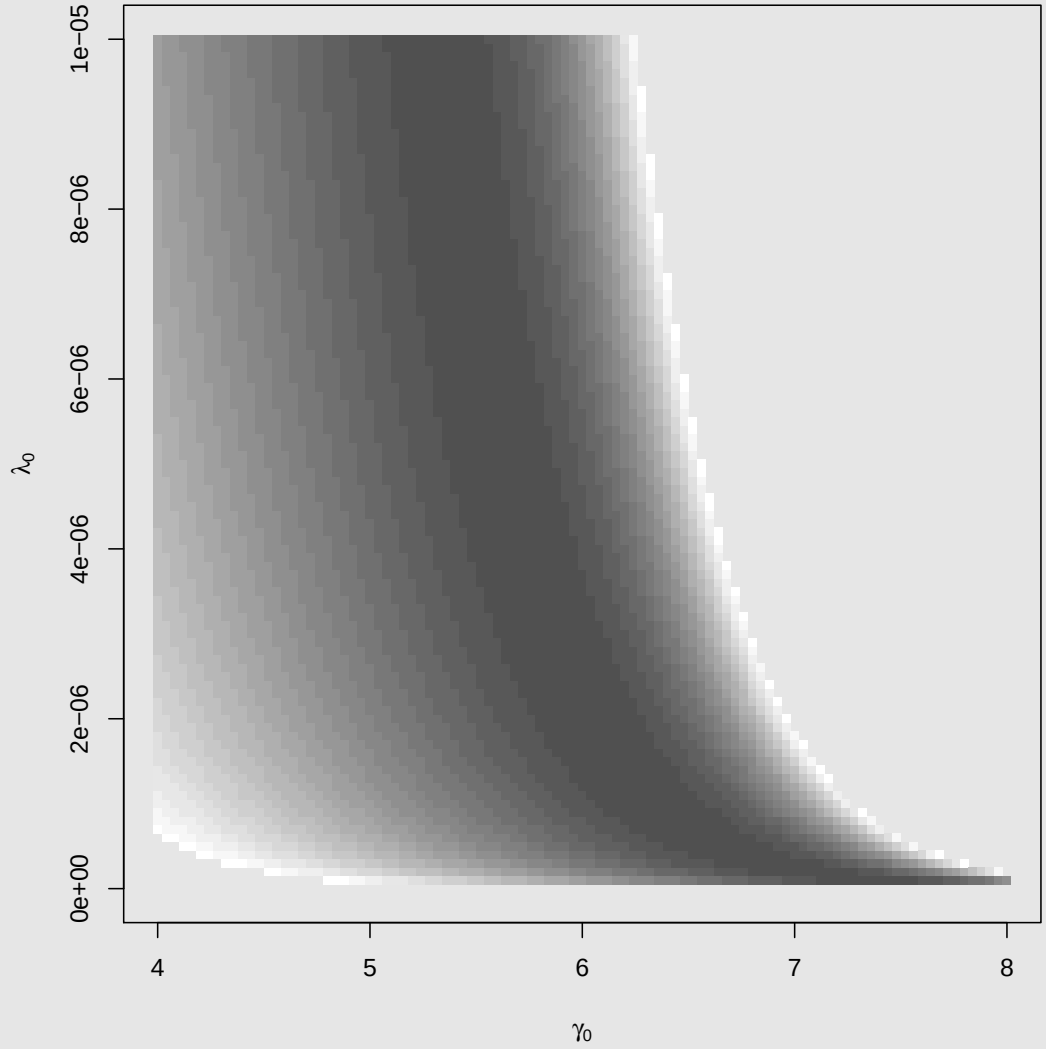
We seek to produce a sample from this joint posterior distribution for data $(n = 15)$.

10.3959 6.2281 6.5331 10.7086 7.6138 8.9423 8.8254 6.1461
7.2988 8.8081 7.5316 8.2238 8.9831 6.4174 9.7648

Gibbs Sampler: example

Example (Weibull example)

Joint posterior (up to proportionality)



Gibbs Sampler: example

Example (Weibull example)

In this case, neither full conditional posterior

$$\pi(\gamma_0|\lambda_0, \boldsymbol{y}) \qquad\qquad \pi(\lambda_0|\gamma_0, \boldsymbol{y})$$

is a standard distribution, and cannot be sampled from easily.

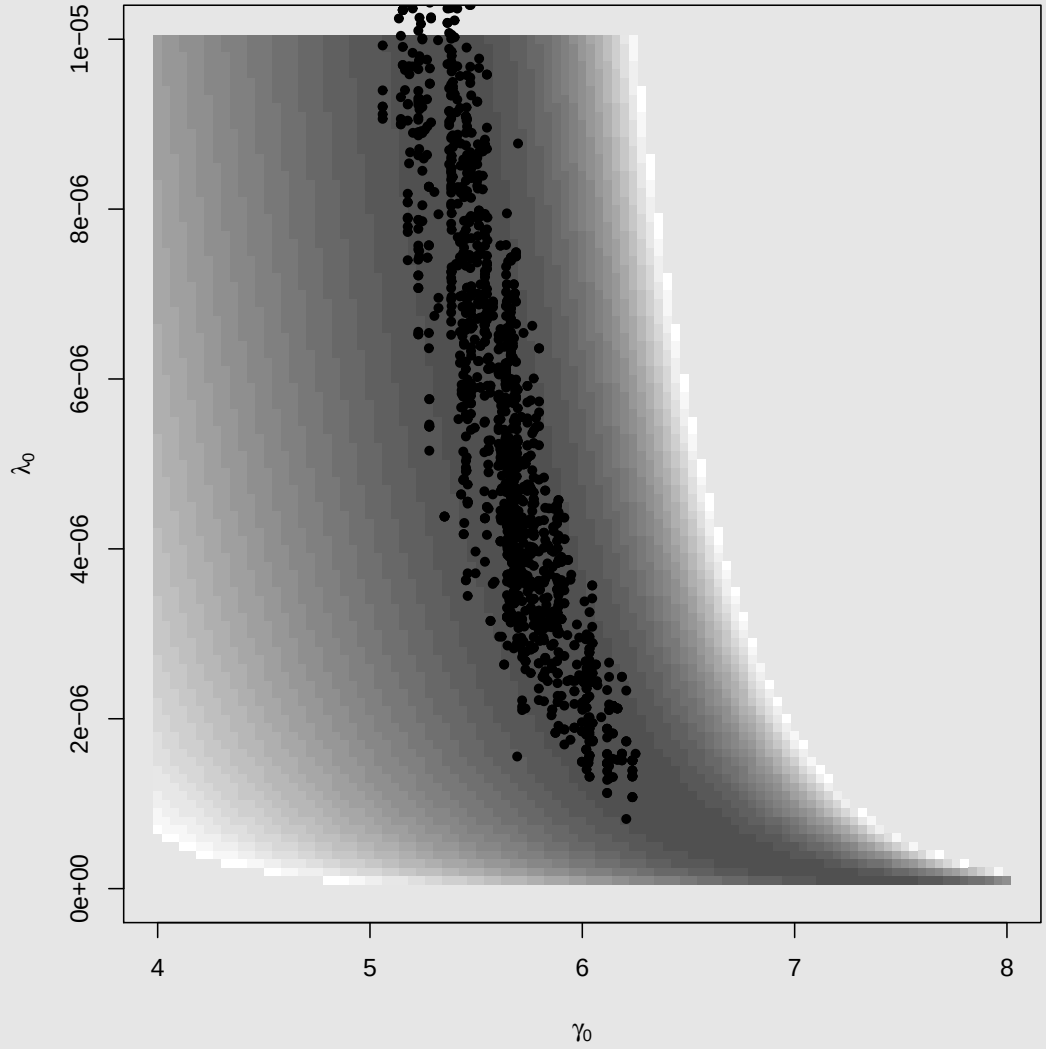
We adopt a *Metropolis-within-Gibbs* strategy, using Metropolis-Hastings accept-reject steps for each parameter and its full conditional.

Specifically, as both parameters are positive, we use the reflected normal proposal distribution from the previous example with $\sigma_q = 1$ for α_0 and $\sigma_q = 10^{-3}$ for λ_0 proposals.

Gibbs Sampler: example

Example (Weibull example)

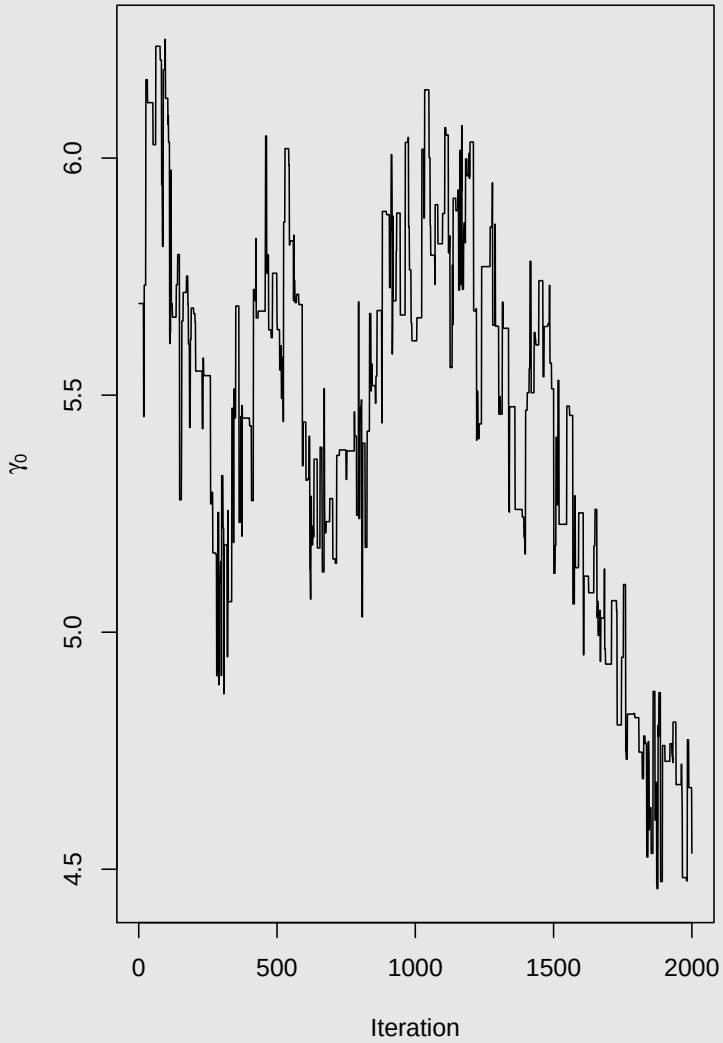
Sample from joint posterior from first 2000 Gibbs sampler steps.

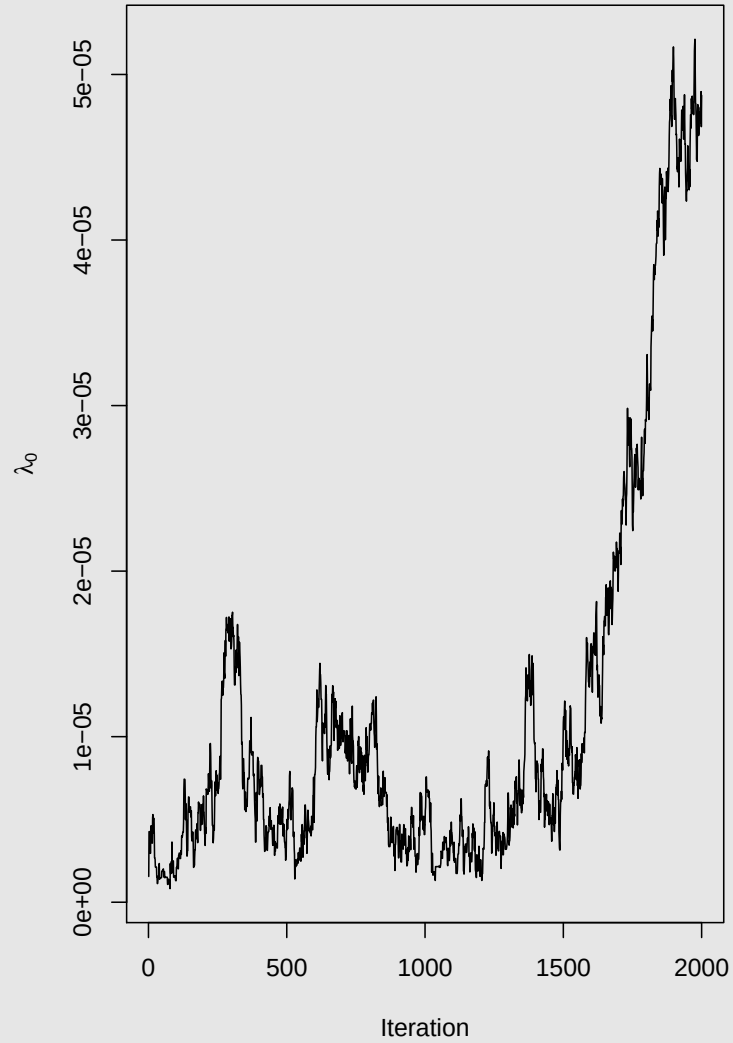


Gibbs Sampler: example

Example (Weibull example)

Trace plots:

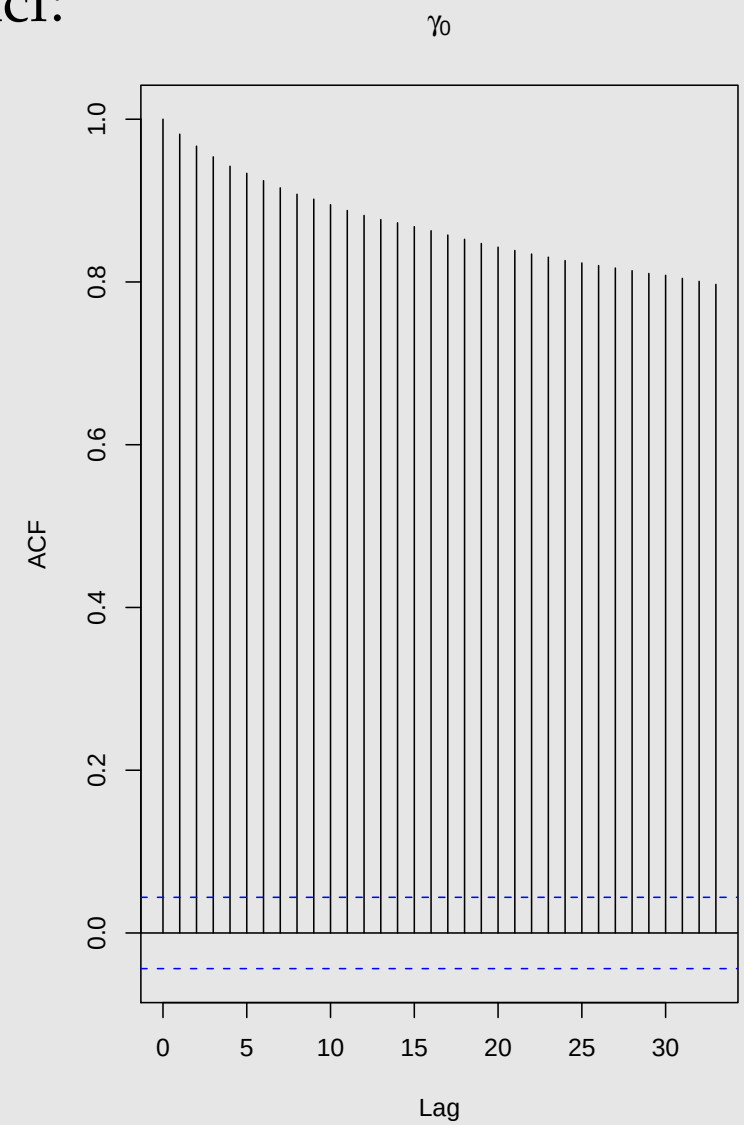


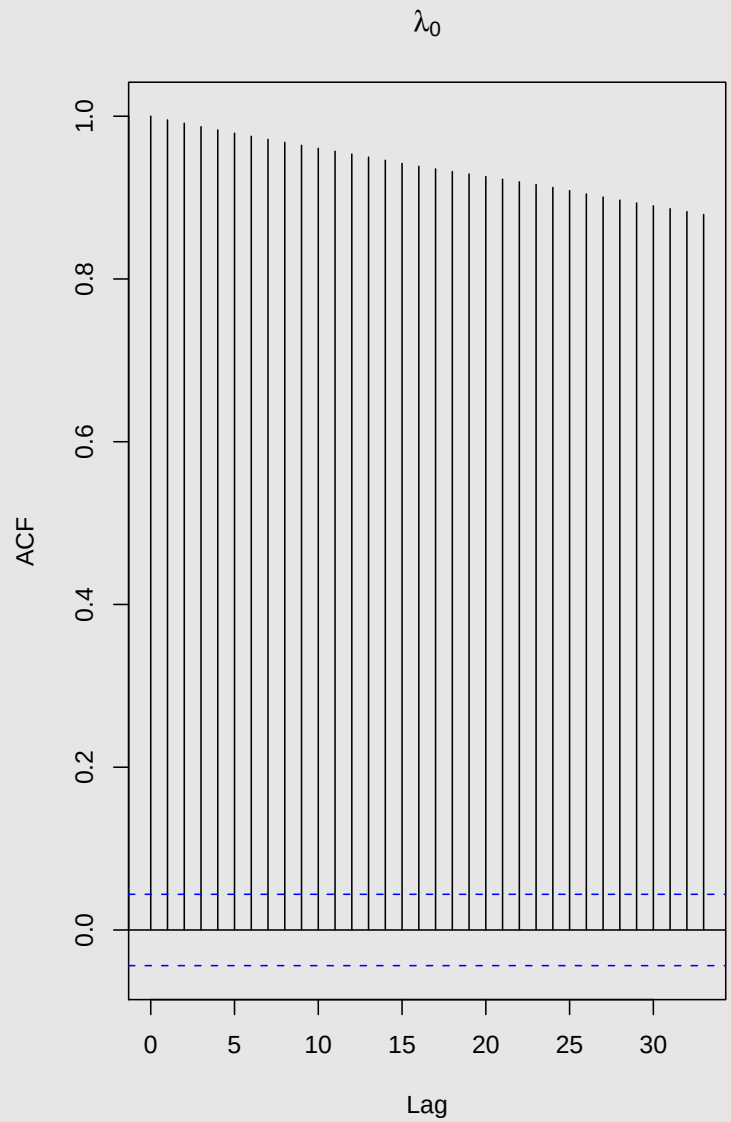


Gibbs Sampler: example

Example (Weibull example)

Acf:





Gibbs Sampler: example

Example (Weibull example)

The posterior correlation here is approximately =0.86; this severely affects the performance of the Gibbs sampler.

An alternative parameterization partially solves this problem. Let $\gamma = \gamma_0$, but

$$\lambda = \left(\frac{1}{\lambda_0}\right)^{1/\gamma} \qquad \lambda_0 = \left(\frac{1}{\lambda}\right)^{\gamma}$$

For this new parameterization, we must remember to include the Jacobian in the prior for the new parameters

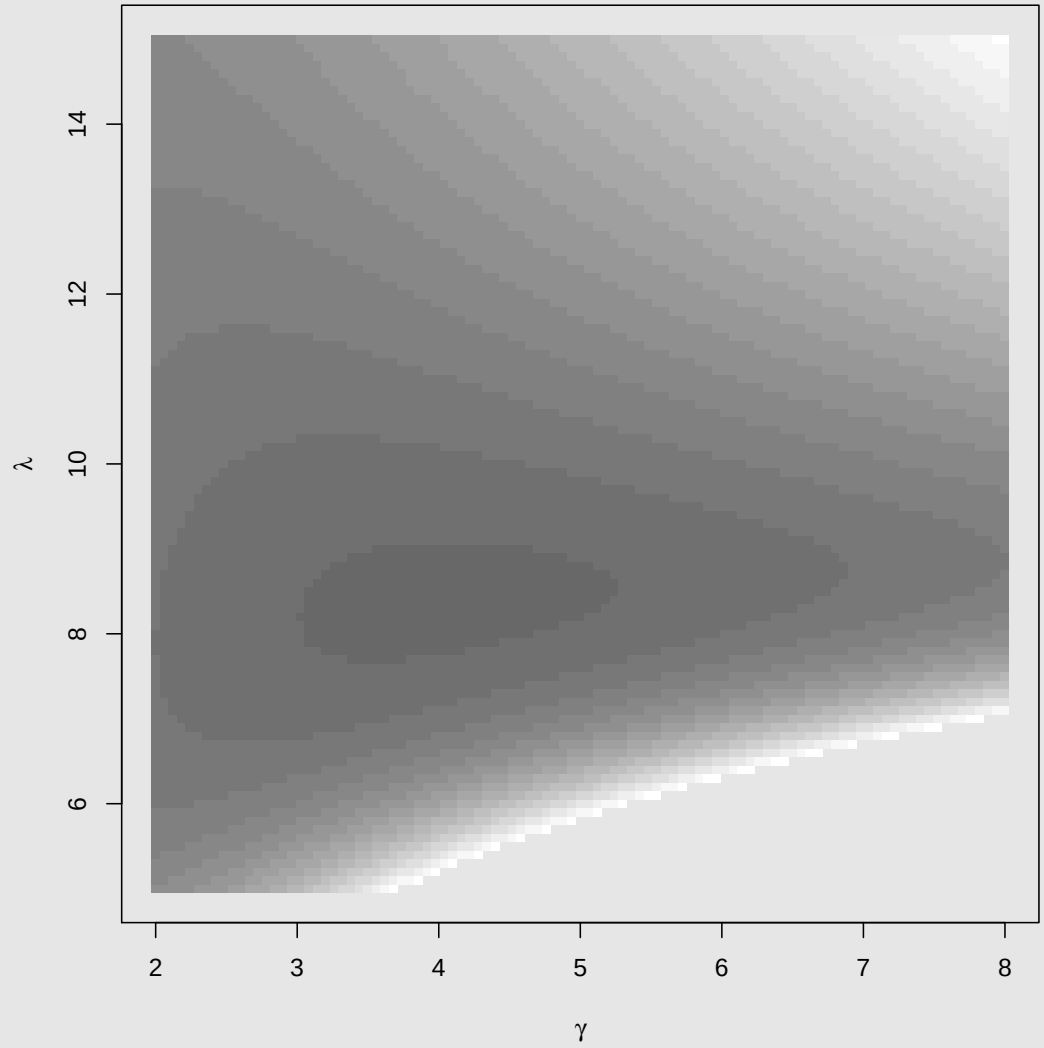
$$p(\gamma, \lambda) = p(\gamma, \lambda_0(\gamma, \lambda))|J(\alpha, \gamma)|$$

We again must use Metropolis-within-Gibbs.

Gibbs Sampler: example

Example (Weibull example)

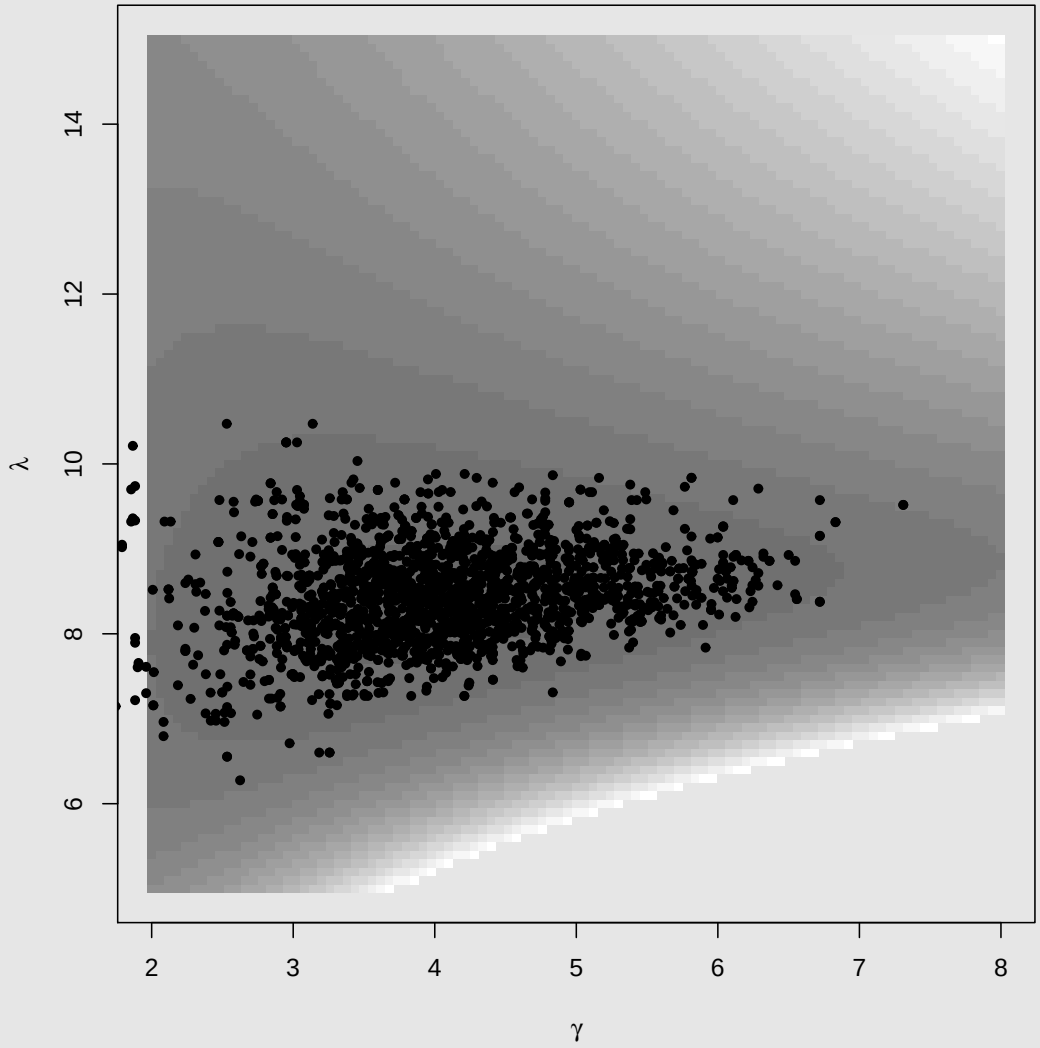
Joint posterior (up to proportionality) for new parameterization



Gibbs Sampler: example

Example (Weibull example)

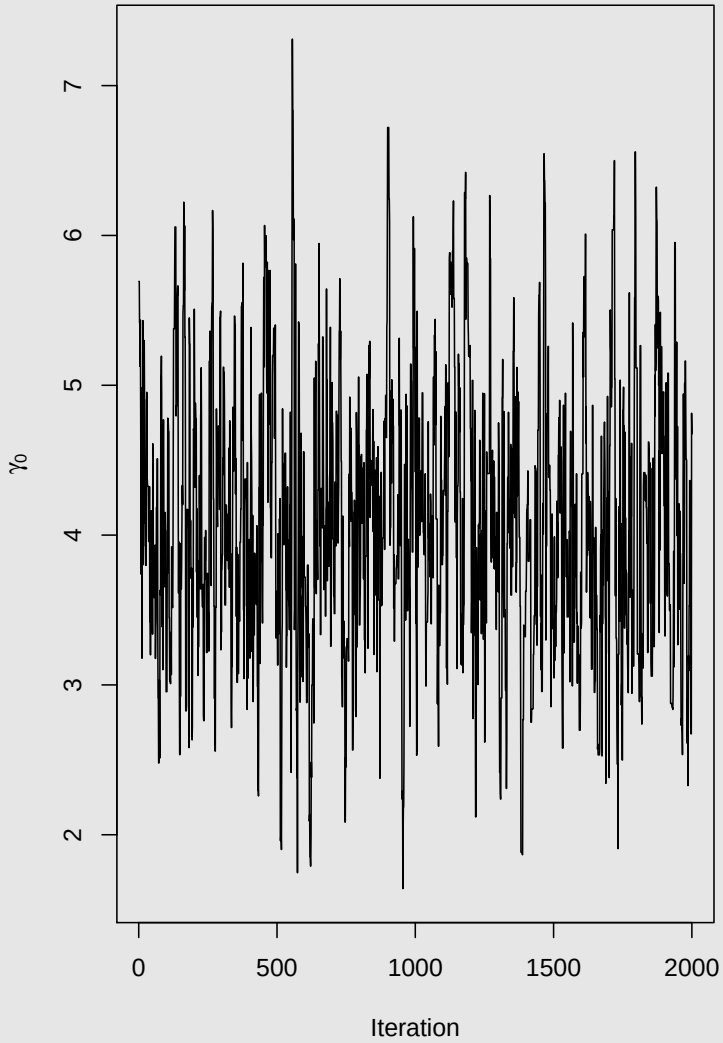
Sample from joint posterior for new parameterization

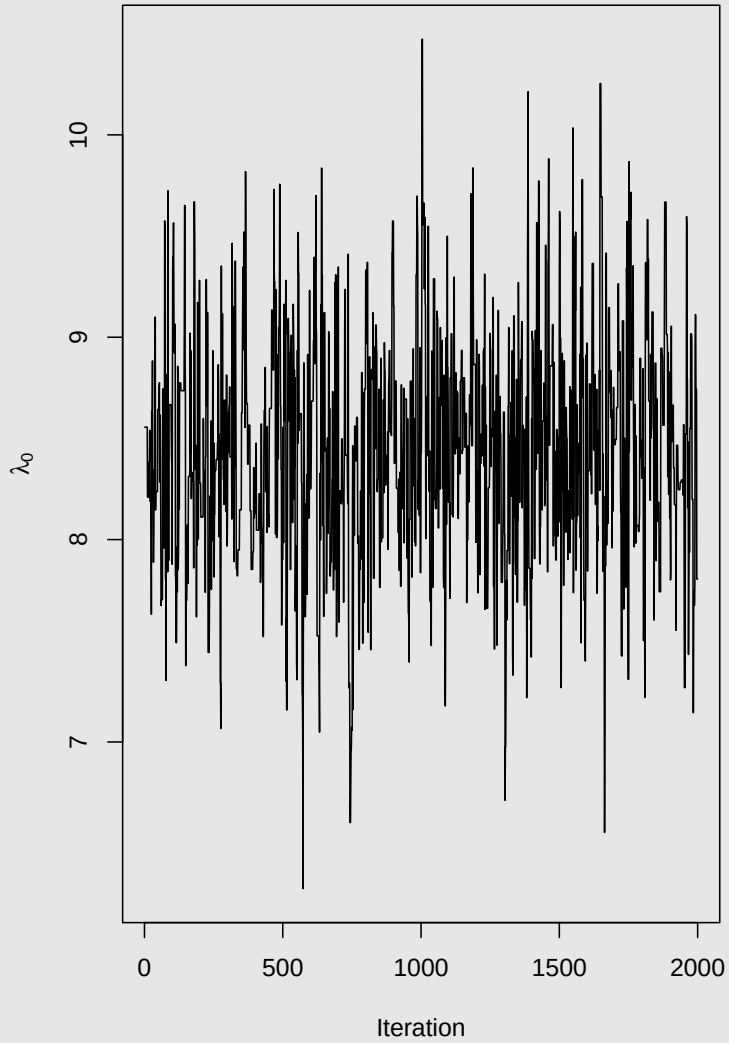


Gibbs Sampler: example

Example (Weibull example)

Trace plots: new parameterization

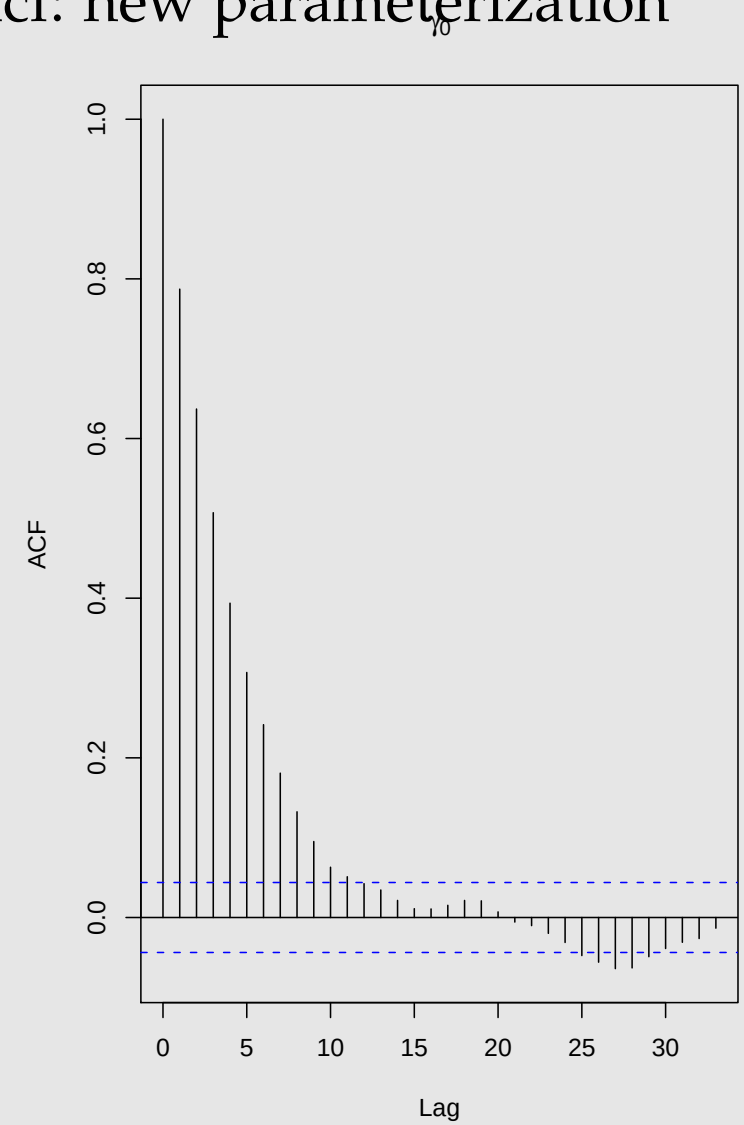


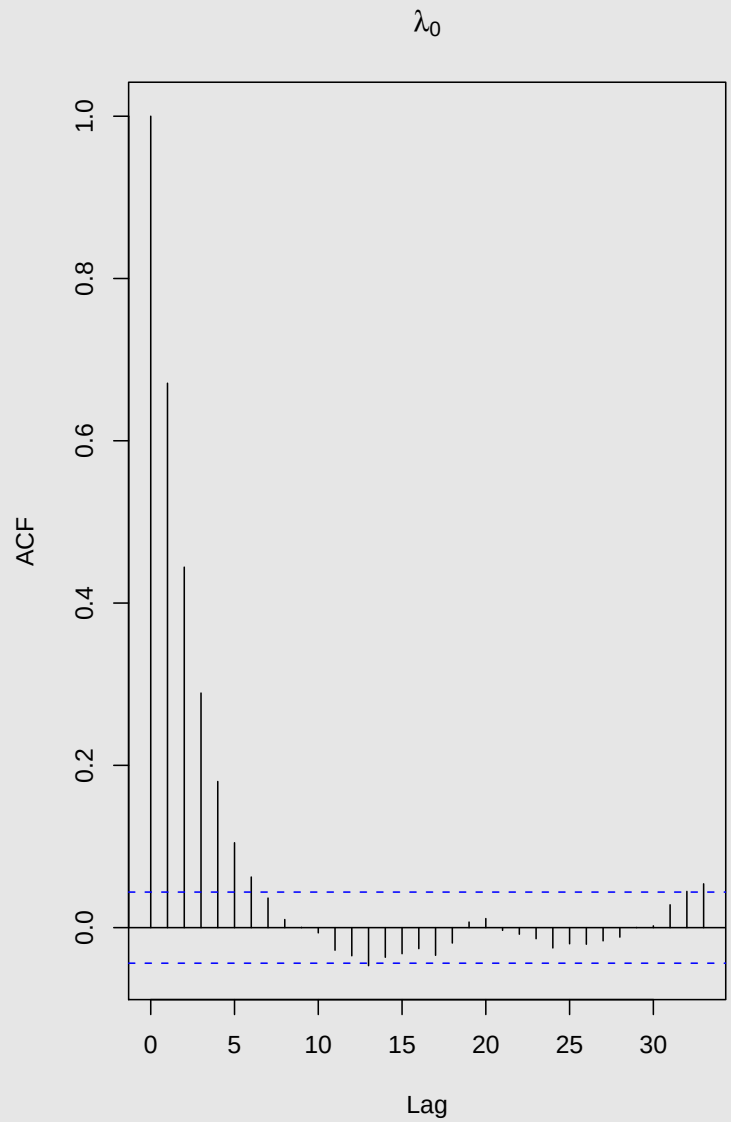


Gibbs Sampler: example

Example (Weibull example)

Acf: new parameterization





Gibbs Sampler: example

Example (Weibull example)

The posterior correlation here is approximately 0.25, and the Gibbs sampler can effectively traverse the parameter space.

Parameter estimates for the new parameters can be obtained from the posterior samples: the mean and 95% credible interval for each parameter is

$$\begin{aligned} \gamma &: 4.079\ (2.455, 6.038) \\ \lambda &: 8.446\ (7.310, 9.593) \end{aligned}$$

It is also possible to obtain posterior summaries for other functions of the posterior parameters.

Gibbs Sampler: example

Example (Weibull example)

For example, the survivor function, $S(y)$, is defined by

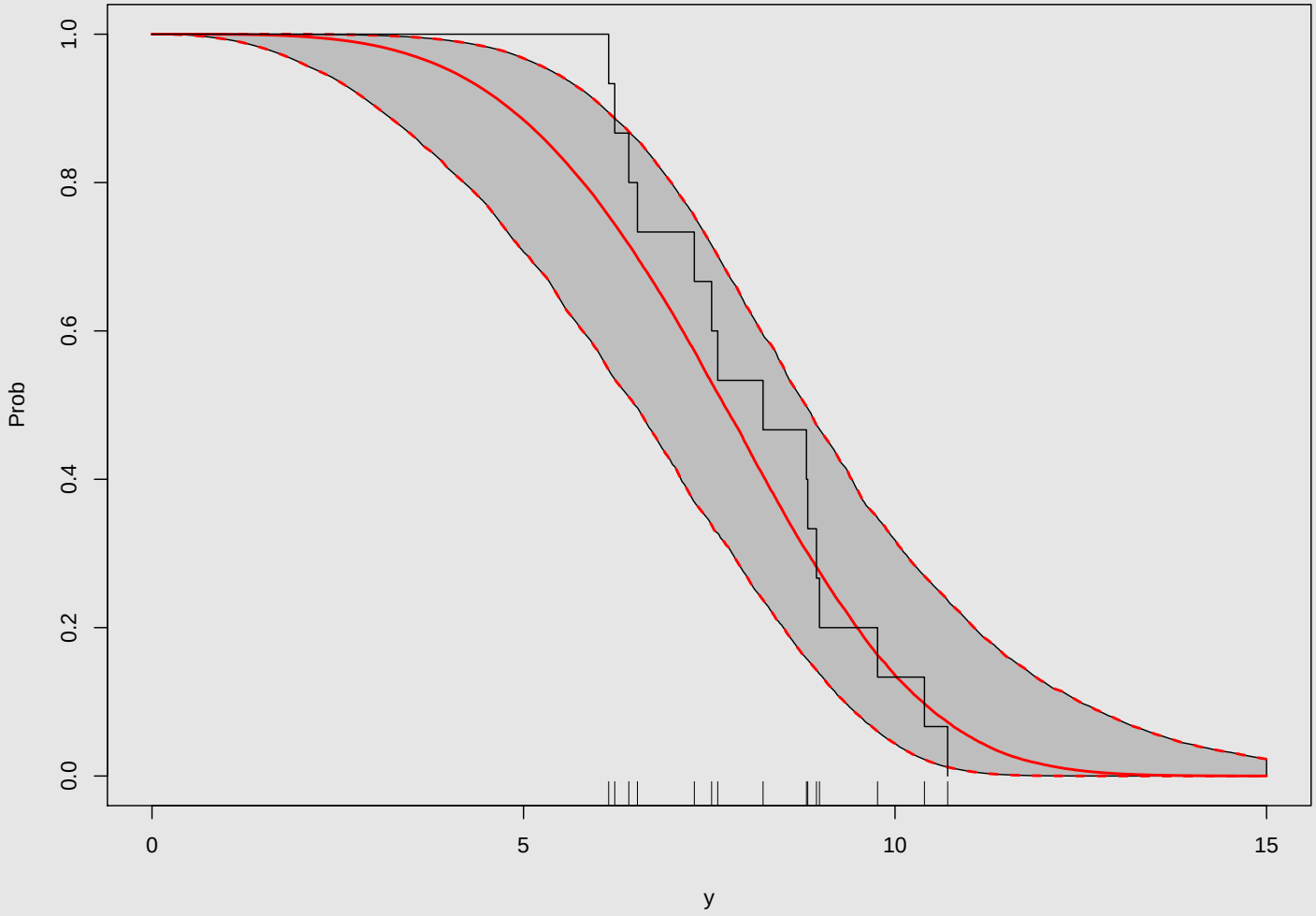
$$S(y) = \mathbb{P}[Y > y] = \exp\{-(y/\lambda)^{\gamma}\}.$$

For each $y \in \mathbb{R}^{+}$, we can compute this function for each sampled pair (γ, λ) obtained from the Gibbs sampler. We can then compute the pointwise credible intervals.

Gibbs Sampler: example

Example (Weibull example)

Posterior survivor function: Bayes estimate and 95 % credible interval (shaded). Solid line is empirical survivor function.



Effective sample size

We seek to measure the adequacy of the collected samples for estimating parameters, in particular, we wish to assess the variance of the estimators.

For an i.i.d. sample of size N, the variance of the Monte Carlo estimator $\widehat{I}_N(g)$ is

$$\frac{\mathbb{V}ar[g(X)]}{N}.$$

However, for a **dependent** sample, the variance is

$$\frac{\mathbb{V}ar[g(X)]}{N_{\text{eff}}}$$

where N_{eff} is the *effective sample size*.

Effective sample size

We have that, for a series of N observations from a dependent stochastic process, the effective sample size is given by

$$N_{\text{eff}} = \frac{N}{1 + 2\sum_{k=1}^{\infty} \rho(k)}$$

where $\rho(k)$ is the true lag-k autocorrelation for the Markov chain. The denominator is termed the *integrated autocorrelation time*.

The true autocorrelations are typically not known, so must be estimated from the data. Most typically this is achieved using spectral methods. The calculation is available in R from the library `coda`, in the function

`effectiveSize`.

Effective sample size

For the Weibull example, with $N = 2000$ in runs shown above

	N_{eff}	
	γ	λ
(γ_0, λ_0) parameterization	8.46	0.00
(γ, λ) parameterization	217.20	393.69

From this we can tell that the second MCMC run, in the (γ, λ) parameterization, has produced much larger effective sample sizes.

The value of zero (highlighted) is the result of apparent lack of convergence.

Rejection sampling for the Weibull example

Note that we could attempt to address the problem of sampling from the posterior distribution in the Weibull problem above by *rejection sampling*. In the (γ, λ) parameterization, we have

$$\pi(\gamma, \lambda|\boldsymbol{y}) \propto L(\gamma, \lambda; \boldsymbol{y}) p(\gamma, \lambda_0(\gamma, \lambda)) |J(\alpha, \gamma)|$$

where $p(.,.)$ is the product of independent $Exponential(0.01)$ priors.

We can use proposal function p_0 which is the product of *Gamma* densities; we choose

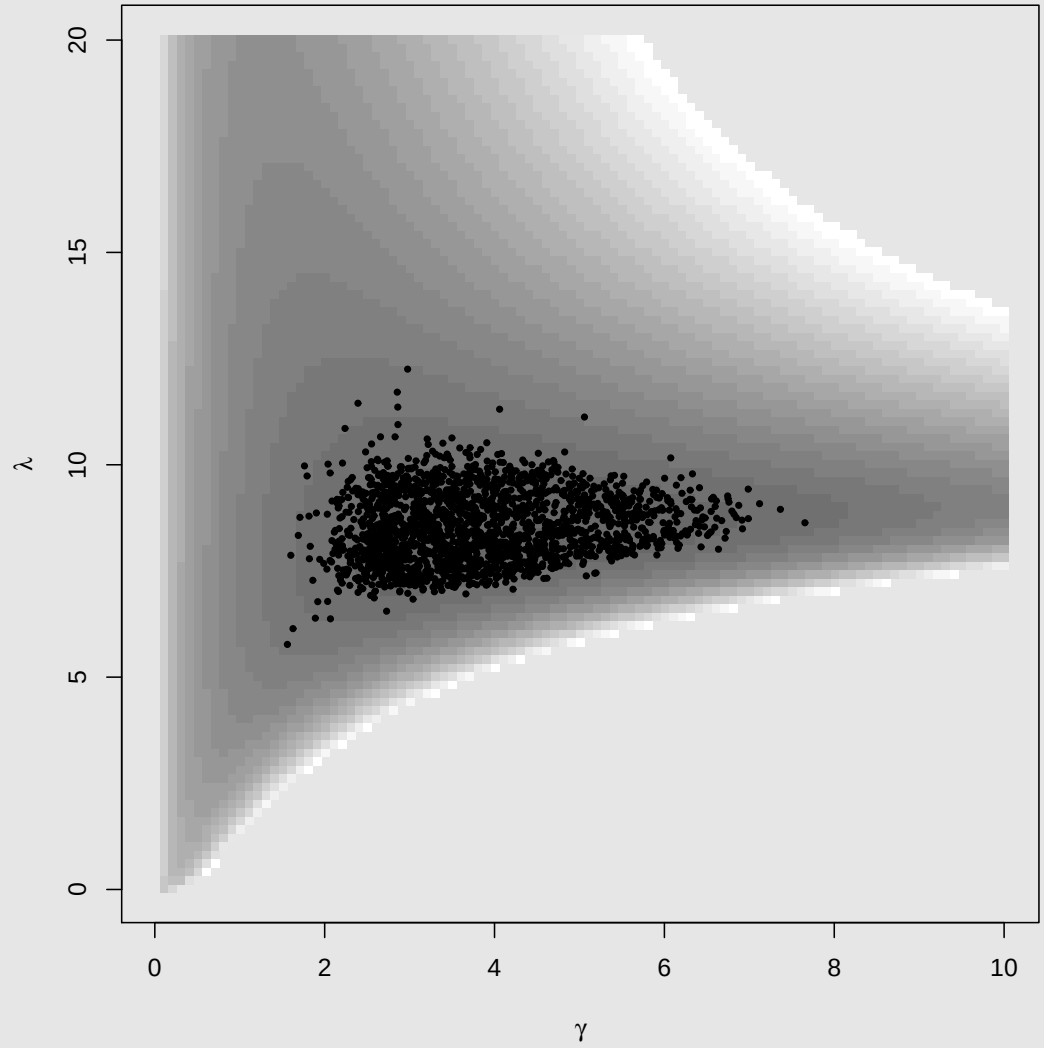
$$Gamma(2, 1/2) \qquad Gamma(4, 2)$$

for proposing (γ, λ), and then use numerical maximization to find M.

Rejection sampling for the Weibull example

Example (Weibull example)

Rejection sampling: acceptance rate is approximately 0.116.



Metropolis-Hastings in higher dimensions

The Metropolis-Hastings algorithm can be used for probability distributions in arbitrary dimension. Note that it is always the case that, for the full conditional distributions, if

- $\boldsymbol{x}_1$ is a sub-vector of the entire vector $\boldsymbol{x}$ of variables, and
- $\boldsymbol{x}_{(1)}$ is $\boldsymbol{x}$ with the components $\boldsymbol{x}_1$ removed,

then

$$\pi(\boldsymbol{x}_1|\boldsymbol{x}_{(1)}) \propto \pi(\boldsymbol{x})$$

as the normalizing constant is $\pi(\boldsymbol{x}_{(1)})$, which does not depend on $\boldsymbol{x}_1$.

For Metropolis-within-Gibbs, knowing $\pi(\boldsymbol{x}_1|\boldsymbol{x}_{(1)})$ up to proportionality is sufficient, as the normalizing constant cancels out in the calculation of the acceptance probability.

Gibbs sampler: example

Example (Ising model)

Recall the Ising model: this is a joint distribution for the collection of binary random variables $\{X_i\}$ placed on a rectangular $(N \times M)$ lattice, with joint mass function

$$\pi_{\mathbf{X}|\beta}(\boldsymbol{x}|\beta) = \frac{\exp\left\{\beta \sum_{i=1}^{NM} \sum_{j \neq i} \mathbb{1}_{x_i}(x_j)\right\}}{Z(\beta)}$$

where $Z(\beta)$ is the normalizing constant. The support of this mass function has

$$2^{NM}$$

elements.

Gibbs sampler: example

Example (Ising model)

The full conditional distribution for the X_i is a discrete distribution on $\{0, 1\}$, with

$$\pi_i(x_i|x_{(i)}) \propto \exp\left\{\beta \sum_{j \in \partial i} \mathbb{1}_{x_j}(x_i)\right\}$$

where ∂i is the neighbourhood of i. This distribution reduces to

$$\mathbb{P}[X_i = 0|X_{(i)} = x_{(i)}] = \frac{e^{\beta n_{i0}}}{e^{\beta n_{i0}} + e^{\beta n_{i1}}}$$

where n_{i0} and n_{i1} are the numbers of neighbours of i that take the values 0 and 1 respectively. This distribution can be sampled easily as part of a Gibbs sampler.